

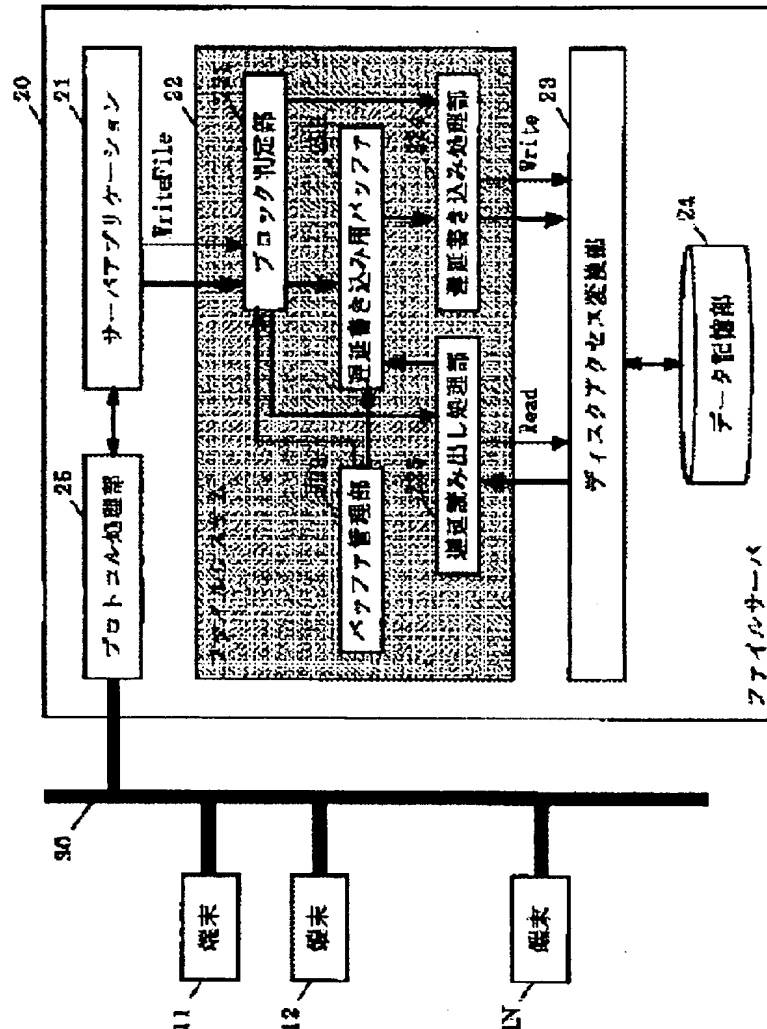
☐ Include

## MicroPatent® PatSearch FullText: Record 1 of 1

Search scope: JP ; Full patent spec.

Years: 2001-2003

Patent/Publication No.: JP2002196961



## Order This Patent

## Family Lookup

**Find Similar**

## Legal Status

Go to first matching text

JP2002196961 A

## FILE SYSTEM

MATSUSHITA ELECTRIC IND CO LTD

Inventor(s) : GOSHIMA YUKIE ; DOI SHINZO ; TSUJI ATSUHIRO

Application No. 2000397721 JP2000397721 JP, Filed 20001227,A1 Published 20020712Published 20020712

**Abstract PROBLEM TO BE SOLVED:** To provide a file system capable of realizing high-speed real-time writing, and utilizing effectively a band width, concerning the file system receiving a file writing request of an optional size.

**SOLUTION:** When receiving the file writing request from a server application 21, a block determination part 221 transfers requested data to a delay writing buffer 223. Then, the block determination part 221 determines whether readout from a data memory part 24 is required or not based on continuity of a writing start position and the position of a block which is a unit for writing in the data memory part 24. A delay readout processing part 225 executes readout processing only when determined as required by the block determination part 221. After the data

is transferred from the block determination part 221 and read out by the delay readout processing part 225, a delay writing processing part 224 writes the block having arranged data in the data memory part 24 at very fixed period.

Int'l Class: G06F01200;

Patents Citing this One: No US, EP, or WO patents/search reports have cited this patent. MicroPatent Reference Number: 000555947

COPYRIGHT: (C) 2002JPO



Home



List

---

For further information, please contact:  
[Technical Support](#) | [Billing](#) | [Sales](#) | [General Information](#)

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2002-196961

(P2002-196961A)

(43) 公開日 平成14年7月12日 (2002.7.12)

(51) Int.Cl.<sup>7</sup>

G 0 6 F 12/00

識別記号

5 1 4

F I

G 0 6 F 12/00

キーワード(参考)

5 1 4 A 5 B 0 8 2

5 1 4 K

審査請求 未請求 請求項の数30 O L (全 28 頁)

(21) 出願番号 特願2000-397721(P2000-397721)

(22) 出願日 平成12年12月27日 (2000. 12. 27)

(71) 出願人 000005821

松下電器産業株式会社

大阪府門真市大字門真1006番地

(72) 発明者 五島 雪絵

大阪府門真市大字門真1006番地 松下電器  
産業株式会社内

(72) 発明者 土居 晋三

大阪府門真市大字門真1006番地 松下電器  
産業株式会社内

(74) 代理人 100098291

弁理士 小笠原 史朗

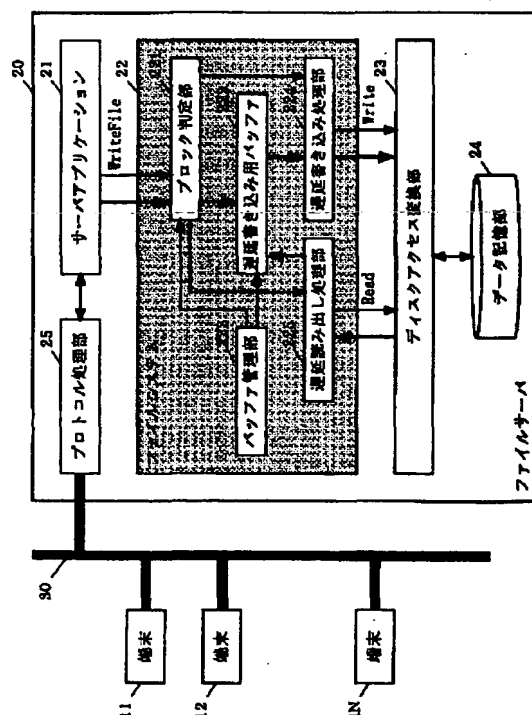
最終頁に続く

(54) 【発明の名称】 ファイルシステム

(57) 【要約】

【課題】 任意のサイズのファイル書き込み要求を受け付けるファイルシステムにおいて、高速なリアルタイム書き込みを実現し、かつ、帯域幅を有効利用できるファイルシステムを提供する。

【解決手段】 サーバアプリケーション21からファイル書き込み要求を受けると、ブロック判定部221は、要求されたデータを遅延書き込み用バッファ223へ転送する。さらに、ブロック判定部221は、書き込み開始位置の連続性と、データ記憶部24に書き込む単位であるブロックの位置とに基づいて、データ記憶部24からの読み出しが必要かを判定する。遅延読み出し処理部225は、ブロック判定部221で必要と判定されたときのみ読み出し処理を行う。遅延書き込み処理部224は、データがブロック判定部221から転送され、遅延読み出し処理部225で読み出された後、データの揃ったブロックをデータ記憶部24へ一定周期毎で書き込む。



**【特許請求の範囲】**

【請求項 1】 アクセス可能な固定長の単位でデータが管理される記憶部に対して、入力されるファイル操作要求に応じたファイル処理を実行するファイルシステムであって、

データが一時的に格納されるバッファ部と、  
入力される任意サイズのデータによるファイル書き込み処理要求に従って、当該データを前記バッファ部へ格納すると共に、前記バッファ部でのファイルデータの格納状態に基づいて、ファイルデータの読み出し対象及び書き込み対象を設定する判定部と、

設定された前記読み出し対象のファイルデータに応じた遅延読み出し処理要求を、ファイルデータの格納位置が、ファイル内の相対的なアドレス情報で示されるファイル形式又は前記記憶部内の絶対的なアドレス情報で示されるデータ形式のいずれかを用いて前記記憶部へ出力することにより、当該読み出し対象のファイルデータを、前記記憶部から前記バッファ部へ読み出す遅延読み出し処理部と、

設定された前記書き込み対象のファイルデータに応じた遅延書き込み処理要求を、前記ファイル形式又は前記データ形式のいずれかを用いて前記記憶部へ出力することにより、当該書き込み対象のファイルデータを、前記バッファ部から前記記憶部へ書き込む遅延書き込み処理部とを備える、ファイルシステム。

【請求項 2】 前記遅延書き込み処理部は、全ファイル又はファイル毎に予め定められた書き込み周期に従って、前記遅延書き込み処理要求を出力することを特徴とする、請求項 1 に記載のファイルシステム。

【請求項 3】 ファイルを所定のブロックデータサイズで分割した領域を示すブロックの単位で、前記バッファ部を管理するバッファ管理部をさらに備え、  
前記バッファ部には、前記バッファ管理部によって、ファイルを構成する複数の前記ブロックにそれぞれ対応する複数のブロック領域が確保され、前記ファイル書き込み処理要求のデータが、ファイル上で対応する当該ブロック領域に格納され、

前記判定部は、所定の判定条件に基づいて、領域の一部にのみデータが格納された前記ブロック領域を前記読み出し対象として設定し、領域の全てにデータが格納された前記ブロック領域を前記書き込み対象として設定し、  
前記遅延読み出し処理部は、前記読み出し対象に設定されたブロック領域に対応するファイルデータの前記遅延読み出し処理要求を出力し、前記記憶部から読み出されたファイルデータの全部又は一部を前記バッファ部に格納し、

前記遅延書き込み処理部は、前記書き込み対象と設定されたブロック領域に対応するファイルデータの前記遅延書き込み処理要求を出力し、当該ブロック領域に格納されているファイルデータの全部又は一部を前記記憶部へ

書き込むことを特徴とする、請求項 1 又は 2 に記載のファイルシステム。

【請求項 4】 前記判定部は、  
最後に入力した前記ファイル書き込み処理要求に従って格納したデータの前記バッファ部上の格納終了位置を、ファイル先頭からのオフセット位置で示される最終アクセス位置として保存し、

新たに入力した前記ファイル書き込み処理要求に従って新たに格納するデータの前記バッファ部上のファイル先頭からのオフセット位置で示される格納開始位置が、前記最終アクセス位置と連続するか否かを判定し、  
位置が連続しないと判定した場合、前記最終アクセス位置を含むブロック領域の全てにデータが格納されていないければ、当該ブロック領域を前記読み出し対象に設定すると共に、当該ブロック領域中のデータが存在しない範囲を遅延読み出しデータ範囲として設定し、

位置が連続しないと判定した場合、前記格納開始位置を含むブロック領域の先頭位置から前記格納開始位置までにデータが存在しない範囲があれば、当該ブロック領域を前記読み出し対象と設定すると共に、当該ブロック領域中の当該範囲を遅延読み出しデータ範囲として設定し、

前記遅延読み出し処理部は、前記遅延読み出し処理要求によって読み出されたファイルデータの全部又は前記遅延読み出しデータ範囲に対応する部分のみを、前記バッファ部の該当する前記ブロック領域又は前記遅延読み出しデータ範囲に格納することを特徴とする、請求項 3 に記載のファイルシステム。

【請求項 5】 前記判定部は、領域の一部にのみデータが格納されたが前記読み出し対象に設定しなかったブロック領域が、予め定められた判定時間を経過した後も前記書き込み対象に設定できない場合には、当該ブロック領域を前記読み出し対象に設定すると共に、当該ブロック領域中のデータが存在しない範囲を遅延読み出しデータ範囲として設定することを特徴とする、請求項 3 又は 4 に記載のファイルシステム。

【請求項 6】 前記記憶部に対するアクセスは、前記固定長の単位に基づく所定のアクセスデータサイズで行われ、

前記ブロックデータサイズは、前記アクセスデータサイズの N 倍 (N は、自然数) であり、  
前記判定部は、前記読み出し対象に設定すべきブロック領域の、データが格納されている範囲のデータサイズが、前記アクセスデータサイズの n 倍 (n は、自然数) であり、かつ、当該範囲の先頭位置が、前記アクセスデータサイズでファイルが分割されたときの区切り位置に一致する場合には、当該ブロック領域を前記書き込み対象に設定し、

前記遅延書き込み処理部は、前記ブロック領域内の未格納範囲へのデータ読み出し処理が行われることなく、前

記範囲のデータのみを前記記憶部の対応する位置へ書き込むことを特徴とする、請求項3～5のいずれかに記載のファイルシステム。

【請求項7】 前記遅延読み出し処理部は、前記読み出し対象に設定されたブロック領域の、前記遅延読み出しデータ範囲が、前記アクセスデータサイズの $m$ 倍（ $m$ は、自然数）であり、かつ、当該遅延読み出しデータ範囲の先頭位置が、前記アクセスデータサイズでファイルが分割されたときの区切り位置に一致する場合には、ファイルデータ内の当該遅延読み出しデータ範囲に該当する部分のみを、前記記憶部から前記バッファ部へ読み出すことを特徴とする、請求項6に記載のファイルシステム。

【請求項8】 前記判定部は、前記バッファ部にすでに格納されている前記ファイル書き込み処理要求の対象となるファイルのデータサイズが、予め定められたバッファデータサイズを超えない場合のみ、新たに入力した前記ファイル書き込み処理要求に従った前記バッファ部へのデータの格納を行うことを特徴とする、請求項3～7のいずれかに記載のファイルシステム。

【請求項9】 前記バッファデータサイズは、前記ブロックデータサイズの $K$ 倍（ $K$ は、自然数）であることを特徴とする、請求項8に記載のファイルシステム。

【請求項10】 アクセス可能な固定長の単位でデータが管理される記憶部に対して、入力されるファイル操作要求に応じたファイル処理を実行するファイルシステムであって、

データが一時的に格納されるバッファ部と、外部から入力される帯域設定要求によって指定される又は予め内部に有する情報から取得する帯域幅に基づいて、全ファイル又はファイル毎に予め定められた書き込み周期を設定する帯域設定部と、

入力される任意サイズのデータによるファイル書き込み処理要求に従って、当該データを前記バッファ部へ格納すると共に、前記バッファ部でのファイルデータの格納状態に基づいて、ファイルデータの読み出し対象及び書き込み対象を設定する判定部と、

設定された前記読み出し対象のファイルデータに応じた遅延読み出し処理要求を、ファイルデータの格納位置が、ファイル内の相対的なアドレス情報で示されるファイル形式又は前記記憶部内の絶対的なアドレス情報で示されるデータ形式のいずれかをを用いて前記記憶部へ出力することにより、当該読み出し対象のファイルデータを、前記記憶部から前記バッファ部へ読み出す遅延読み出し処理部と、

前記書き込み周期に従って、設定された前記書き込み対象のファイルデータに応じた遅延書き込み処理要求を、前記ファイル形式又は前記データ形式のいずれかをを用いて前記記憶部へ出力することにより、当該書き込み対象のファイルデータを、前記バッファ部から前記記憶部へ

書き込む遅延書き込み処理部とを備える、ファイルシステム。

【請求項11】 ファイルを所定のブロックデータサイズで分割した領域を示すブロックの単位で、前記バッファ部を管理するバッファ管理部をさらに備え、前記バッファ部には、前記バッファ管理部によって、ファイルを構成する複数の前記ブロックにそれぞれ対応する複数のブロック領域が確保され、前記ファイル書き込み処理要求のデータが、ファイル上で対応する当該ブロック領域に格納され、

前記判定部は、所定の判定条件に基づいて、領域の一部にのみデータが格納された前記ブロック領域を前記読み出し対象として設定し、領域の全てにデータが格納された前記ブロック領域を前記書き込み対象として設定し、前記遅延読み出し処理部は、前記読み出し対象に設定されたブロック領域に対応するファイルデータの遅延読み出し処理要求を出力し、前記記憶部から読み出されたファイルデータの全部又は一部を前記バッファ部に格納し、

前記遅延書き込み処理部は、前記書き込み対象と設定されたブロック領域に対応するファイルデータの遅延書き込み処理要求を出力し、当該ブロック領域に格納されているファイルデータの全部又は一部を前記記憶部へ書き込むことを特徴とする、請求項10に記載のファイルシステム。

【請求項12】 前記判定部は、最後に入力した前記ファイル書き込み処理要求に従って格納したデータの遅延書き込み処理部上の格納終了位置を、ファイル先頭からのオフセット位置で示される最終アクセス位置として保存し、

新たに入力した前記ファイル書き込み処理要求に従って新たに格納するデータの遅延書き込み処理部上のファイル先頭からのオフセット位置で示される格納開始位置が、前記最終アクセス位置と連続するか否かを判定し、

位置が連続しないと判定した場合、前記最終アクセス位置を含むブロック領域の全てにデータが格納されていなければ、当該ブロック領域を前記読み出し対象に設定すると共に、当該ブロック領域中のデータが存在しない範囲を遅延読み出しデータ範囲として設定し、

位置が連続しないと判定した場合、前記格納開始位置を含むブロック領域の先頭位置から前記格納開始位置までにデータが存在しない範囲があれば、当該ブロック領域を前記読み出し対象と設定すると共に、当該ブロック領域中の当該範囲を遅延読み出しデータ範囲として設定し、

前記遅延読み出し処理部は、前記遅延読み出し処理要求によって読み出されたファイルデータの全部又は前記遅延読み出しデータ範囲に対応する部分のみを、前記バッファ部の該当する前記ブロック領域又は前記遅延読み出しデータ範囲に格納することを特徴とする、請求項11

に記載のファイルシステム。

【請求項 13】 前記帯域設定部は、前記帯域設定要求によって指定される又は前記情報から取得する帯域幅に基づいて、全ファイル又はファイル毎に判定時間を設定し、

前記判定部は、領域の一部にのみデータが格納されたが前記読み出し対象に設定しなかったブロック領域が、前記判定時間を経過した後も前記書き込み対象に設定できない場合には、当該ブロック領域を前記読み出し対象に設定すると共に、当該ブロック領域中のデータが存在しない範囲を遅延読み出しデータ範囲として設定することを特徴とする、請求項 11 又は 12 に記載のファイルシステム。

【請求項 14】 前記記憶部に対するアクセスは、前記固定長の単位に基づく所定のアクセスデータサイズで行われ、

前記ブロックデータサイズは、前記アクセスデータサイズの  $N$  倍 ( $N$  は、自然数) であり、

前記判定部は、前記読み出し対象に設定すべきブロック領域の、データが格納されている範囲のデータサイズが、前記アクセスデータサイズの  $n$  倍 ( $n$  は、自然数) であり、かつ、当該範囲の先頭位置が、前記アクセスデータサイズでファイルが分割されたときの区切り位置に一致する場合には、当該ブロック領域を前記書き込み対象に設定し、

前記遅延書き込み処理部は、前記ブロック領域内の未格納範囲へのデータ読み出し処理が行われることなく、前記範囲のデータのみを前記記憶部の対応する位置へ書き込むことを特徴とする、請求項 11～13 のいずれかに記載のファイルシステム。

【請求項 15】 前記遅延読み出し処理部は、前記読み出し対象に設定されたブロック領域の、前記遅延読み出しデータ範囲が、前記アクセスデータサイズの  $m$  倍 ( $m$  は、自然数) であり、かつ、当該遅延読み出しデータ範囲の先頭位置が、前記アクセスデータサイズでファイルが分割されたときの区切り位置に一致する場合には、ファイルデータ内の当該遅延読み出しデータ範囲に該当する部分のみを、前記記憶部から前記バッファ部へ読み出すことを特徴とする、請求項 14 に記載のファイルシステム。

【請求項 16】 前記帯域設定部は、前記帯域設定要求によって指定される又は前記情報から取得する帯域幅に基づいて、バッファデータサイズを設定し、

前記判定部は、前記バッファ部にすでに格納されている前記ファイル書き込み処理要求の対象となるファイルのデータサイズが、前記バッファデータサイズを超えない場合のみ、新たに入力した前記ファイル書き込み処理要求に従った前記バッファ部へのデータの格納を行うことを特徴とする、請求項 11～15 のいずれかに記載のファイルシステム。

【請求項 17】 前記帯域設定部は、前記バッファデータサイズを、前記ブロックデータサイズの  $K$  倍 ( $K$  は、自然数) に設定することを特徴とする、請求項 16 に記載のファイルシステム。

【請求項 18】 アクセス可能な固定長の単位でデータが管理される記憶部に対して、入力されるファイル操作要求に応じたファイル処理を実行するファイルシステムであって、

データが一時的に格納されるバッファ部と、

入力される所定の固定サイズのデータによるファイル書き込み処理要求に従って、当該データを前記バッファ部へ格納すると共に、前記バッファ部でのファイルデータの格納状態に基づいて、ファイルデータの書き込み対象を設定する判定部と、

設定された前記書き込み対象のファイルデータに応じた遅延書き込み処理要求を、ファイルデータの格納位置がファイル内の相対的なアドレス情報で示されるファイル形式を用いて前記記憶部へ出力することにより、当該書き込み対象のファイルデータを、前記バッファ部から前記記憶部へ書き込む遅延書き込み処理部とを備える、ファイルシステム。

【請求項 19】 前記遅延書き込み処理部は、全ファイル又はファイル毎に予め定められた書き込み周期に従って、前記遅延書き込み処理要求を出力することを特徴とする、請求項 18 に記載のファイルシステム。

【請求項 20】 ファイルを所定のブロックデータサイズで分割した領域を示すブロックの単位で、前記バッファ部を管理するバッファ管理部をさらに備え、

前記バッファ部には、前記バッファ管理部によって、ファイルを構成する複数の前記ブロックにそれぞれ対応する複数のブロック領域が確保され、前記ファイル書き込み処理要求のデータが、ファイル上で対応する当該ブロック領域に格納され、

前記判定部は、所定の判定条件に基づいて、領域の全てにデータが格納された前記ブロック領域を前記書き込み対象として設定し、

前記遅延書き込み処理部は、前記書き込み対象と設定されたブロック領域に対応するファイルデータの遅延書き込み処理要求を出力し、当該ブロック領域に格納されているファイルデータの全部又は一部を前記記憶部へ書き込むことを特徴とする、請求項 18 又は 19 に記載のファイルシステム。

【請求項 21】 前記判定部は、

最後に入力した前記ファイル書き込み処理要求に従って格納したデータの遅延書き込み部上の格納終了位置を、ファイル先頭からのオフセット位置で示される最終アクセス位置として保存し、

新たに入力した前記ファイル書き込み処理要求に従って新たに格納するデータの遅延書き込み部上のファイル先頭からのオフセット位置で示される格納開始位置が、前

前記端末は、請求項 1～28 のいずれかに記載のファイ

ルシステムを備え、前記ファイル処理の要求を、ファイルデータの格納位置がファイル内の相対的なアドレス情報で示されるファイル形式を用いて前記ファイルサーバへ出力し、

前記ファイルサーバは、アクセス可能な固定長の単位でデータが管理される記憶部を備え、前記電気通信回線を介して前記端末から前記ファイル処理の要求を入力し、当該要求に応じた処理を当該記憶部に対して行うことを特徴とする、ネットワークシステム。

#### 【発明の詳細な説明】

##### 【0001】

【発明の属する技術分野】本発明は、ファイルシステムに関し、より特定的には、ファイルを蓄積するサーバとファイル操作を行う複数のクライアントとが、ネットワークを介して接続された環境において用いられ、各クライアントがサーバに要求するファイル操作を、サーバ内でリアルタイムかつローカルに実行するためのローカル・ファイルシステム、及びクライアント内でリアルタイムかつリモートで実行するためのリモート・ファイルシステムに関する。

##### 【0002】

【従来の技術】従来、ビデオサーバ装置と端末（クライアント）とをネットワークで接続し、映像や音声をリアルタイムでユーザに提供するようなシステムが存在する。このシステムでは、ビデオサーバ装置が備える大容量の記憶装置に様々な映像や音声のデータが予め記録されており、ビデオサーバ装置が、端末から受けた要求に応じて記憶装置からデータを読み出し、端末へ映像・音声ストリームとして出力する。また、映像や音声のデータをファイルとして扱えるようにファイルサーバ装置に格納し、上記ビデオサーバ装置と同様に、要求に応じて映像や音声を端末へ提供するシステムも提案されている。このように映像や音声のデータをファイルとして扱うと、データをアクセスする際のインタフェース（ファイル読み出し、ファイル書き込み）が、従来のファイルサーバ装置（主に文字情報や静止画像等のデータを扱うサーバ装置）と共通にできる。このため、ファイルサーバ装置を両方の用途に使用することが可能となり、ソフトウェア・ハードウェア資源を有効に活用できる。

【0003】ファイルサーバ装置において、データをファイルとして扱えるようにするのがファイルシステムである。ファイルシステムは、アプリケーションからの要求に応じ、記録装置（例えば、HDD）に対してファイル作成やファイルの読み出し／書き込み等のファイル操作を行うものである。一般的には、ファイルシステムは、オペレーティングシステム（OS）の一機能としてソフトウェアの形態で提供される。特に、単体のパーソナルコンピュータやサーバ機器が、自己に接続されている記録装置に対してファイル操作を行うようなシステムを「ローカル・ファイルシステム」と呼び、クライアン

ト・サーバ・システム等において、クライアント側が、ネットワークを通じてサーバ側の記録装置に対してファイル操作を行うようなシステムを「リモート・ファイルシステム」と呼ぶ。以下、従来のファイルシステムで行われるファイル操作の内、ファイル書き込み処理を幾つか説明する。このファイル書き込み処理は、ファイルの新規作成時又は既存ファイルの内容の変更や追加時に行われるが、以下の説明では、後者の場合を説明する。

【0004】まず、主に文字情報等のデータを扱うファイルサーバを備えたファイルシステムで行われる書き込み処理について説明する（第1の従来例）。この第1の従来例による処理では、ファイルシステムがアプリケーションから書き込み要求を受け取ると、ファイルシステム内部に有するキャッシュメモリに、該当する既存ファイルのデータがあるか否かが判断される。もし、キャッシュメモリにデータが無い場合は、該当するデータが記憶装置からキャッシュメモリに読み出される。このような処理が行われるのは、アプリケーションから任意のサイズで書き込み要求がされるのに対し、記憶装置がデータ読み書き要求を受け付けることのできる最小サイズが通常予め定められているため、ファイルシステムが、この最小サイズの単位（セクタサイズ＝512バイト等）でしかデータのアクセスができないからである。従って、ファイルシステムは、書き込み要求がたとえ1バイトだけであっても、この1バイトのデータだけを記憶装置へ書き込むことができない。従って、ファイルシステムは、記憶装置から最小サイズ分のデータをキャッシュメモリに一度読み出し、1バイトの内容を変更した後、再びキャッシュメモリの最小サイズ分のデータを記憶装置に書き込む必要が生じるのである。

【0005】しかし、上記第1の従来例による処理では、該当するデータが、キャッシュメモリに存在したときにはアプリケーションへの応答が速くなるが、存在しなかったときには、記憶装置からの読み出しという通常のメモリ間でのデータ転送に比べて時間のかかる処理が発生するため、アプリケーションへの応答が遅くなる。この遅れは、文字等のデータを書き込む場合には問題にならないが、映像や音声等のリアルタイムな読み書きが求められるデータにとっては問題になる。例えば、リアルタイムエンコーダ搭載の端末で動画データを録画するような場合、キャッシュメモリに存在するデータについては、必要とされる一定の帯域幅（データ転送速度）での書き込みが可能であるが、キャッシュメモリに存在しないデータについては、記憶装置からの読み出しが行われるため、必要とされる一定の帯域幅での書き込みができなくなる。その結果、データ転送の途切れによるエラー処理で録画がストップしたり、録画した結果にデータ欠落が生じる等の支障がでる。

【0006】一方、ファイル書き込み処理時に、記憶装置からデータ読み出しを行わないファイルシステムも提



案されている（第2の従来例）。この第2の従来例の技術は、映像や音声等の高速かつリアルタイムな転送が必要なデータを対象にしたファイルサーバで用いられる手法であり、アプリケーションからファイルシステムへ要求するファイル書き込みの単位を、任意のサイズではなく、記憶装置のアクセス可能な最小サイズの単位（又は、その倍数の単位）に限定している。従って、書き込み要求が常に最小サイズの単位となるので（上述のように1バイトで要求されることがない）、ファイルシステムは、記憶装置からのデータ読み出しを行う必要がなくなるのである。

【0007】しかし、上記第2の従来例の手法では、記憶装置からのデータ読み出しは必要ないが、文字情報等をデータとする従来のファイルをアクセスしようとする、ファイルアクセスの単位が限定されている（常に固定サイズでの読み書きしかできない）ので、扱い難くなる。また、現在普及しているアプリケーションは、通常、文字情報等のデータを扱うため任意のサイズでの読み書きをするように作られているので、上記第2の従来例のファイルシステムでは、既存のアプリケーションを動作させることができず、ソフトウェア資源の有効活用もできない。

【0008】さらに、ファイルシステムへのアクセス単位が任意のサイズであり、かつ、上記第1の従来例で生じるデータ読み出しによる遅延を抑える方法が、特開平10-171689号公報で開示されている（第3の従来例）。この第3の従来例による方法では、ファイルのデータを一定周期でキャッシュメモリに読み出す機構を設け、データをキャッシュメモリに先読みしておくことで、書き込み要求されたデータをキャッシュメモリに存在させるようにしている。この処理により、記憶装置からのデータ読み出しにかかる処理時間が、書き込み要求したアプリケーションへの応答時間には含まれなくなるため、第3の従来例による方法では、一定の帯域幅でのデータ書き込みが可能になる。

【0009】

【発明が解決しようとする課題】上記のように、第3の従来例による方法では、一定周期毎にデータの先読みを行い、書き込み要求されたデータを高い確率でキャッシュにヒットさせる（存在させる）ようにしている。しかしながら、一定周期毎にデータの先読みを行う場合、実際にサーバに書き込む帯域とは別に、データ先読み用としてほぼ同等の帯域が必要となり、通常の場合に比べ約2倍の書き込み帯域を占有することになる。すなわち、サーバの記憶装置が実際に読み書きできる転送速度に比べ、サーバが端末に提供可能な転送速度が、かなり低くなってしまいう問題がある。

【0010】例えば、サーバに接続されているハードディスクの最大データ転送速度（データ読み書きの合計）が240Mbpsの場合、60Mbpsのファイル（例

えば、DVCPRO50フォーマットのファイル）を、通常4つ同時に読み出しすることができる。しかし、上記処理を行う場合、1ファイルに毎に先読み用帯域60Mbpsと書き込み用帯域60Mbpsとが必要となるので、同時に2ファイルだけしか書き込み処理できないことになる。このことは、全端末が必要とするデータ転送速度に比べてサーバの性能が非常に高い場合には、あまり問題にならないかもしれないが、サーバの性能を最大限に使用するような端末数・帯域幅の条件では、大きな問題になる。

【0011】さらに、リアルタイム書き込みが求められるシステムの用途としては、例えば、リアルタイムエンコーダ搭載の端末で、動画データをリアルタイムにサーバの記憶装置に書き込む場合が多い。この用途では、ファイル内の位置を毎回変えながら部分書き換える（不連続アクセス）場合より、長時間分のデータを一気にシークエンシャルに書き込む（連続アクセス）場合が多い。このため、上述した先読み処理でせっかくキャッシュしておいたデータも、後から書き込み要求されるデータでほとんど上書きされてしまうことになる。従って、本来読み出しが必要とされるデータ範囲（書き込み要求のデータが存在しない範囲）に比べ、先読みのデータ範囲がかなり大きくなり、非効率である。

【0012】それ故、本発明の目的は、任意のサイズのファイル書き込み要求を受け付けるファイルシステムにおいて、高速なリアルタイム書き込みを実現し、かつ、帯域幅を有効利用できるファイルシステムを提供することである。また、本発明のさらなる目的は、書き込み要求に対する応答遅延が、記憶装置からの読み出しによって遅れることなく、動画等のリアルタイム転送が必要なデータの書き込みを、常時支障なく行うことのできるリアルタイムなファイルシステムを提供することである。

【0013】

【課題を解決するための手段および発明の効果】第1の発明は、アクセス可能な固定長の単位でデータが管理される記憶部に対して、入力されるファイル操作要求に応じたファイル処理を実行するファイルシステムであって、データが一時的に格納されるバッファ部と、入力される任意サイズのデータによるファイル書き込み処理要求に従って、当該データをバッファ部へ格納すると共に、バッファ部でのファイルデータの格納状態に基づいて、ファイルデータの読み出し対象及び書き込み対象を設定する判定部と、設定された読み出し対象のファイルデータに応じた遅延読み出し処理要求を、ファイルデータの格納位置が、ファイル内の相対的なアドレス情報で示されるファイル形式又は記憶部内の絶対的なアドレス情報で示されるデータ形式のいずれかをを用いて記憶部へ出力することにより、当該読み出し対象のファイルデータを、記憶部からバッファ部へ読み出す遅延読み出し処理部と、設定された書き込み対象のファイルデータに応

じた遅延書き込み処理要求を、ファイル形式又はデータ形式のいずれかをを用いて記憶部へ出力することにより、当該書き込み対象のファイルデータを、バッファ部から記憶部へ書き込む遅延書き込み処理部とを備える。

【0014】上記のように、第1の発明によれば、記憶部へファイルデータを書き込むにあたり、バッファ部でのファイルデータの格納状態に基づいて、読み出しが必要なデータだけを記憶部から読み出す。これにより、任意サイズのファイル書き込み処理要求に対しても、リアルタイムにファイルデータの書き込み処理を行うことができる。また、常にキャッシュメモリにデータを先読みしておく従来の方式に比べて読み出し処理の無駄がなく、帯域幅の有効利用ができる。

【0015】第2の発明は、第1の発明に従属する発明であって、遅延書き込み処理部は、全ファイル又はファイル毎に予め定められた書き込み周期に従って、遅延書き込み処理要求を出力することを特徴とする。

【0016】上記のように、第2の発明によれば、データ転送速度が異なるファイルに対しても帯域幅を無駄にせず、適切な帯域幅での書き込みが行える。

【0017】第3の発明は、第1及び第2の発明に従属する発明であって、ファイルを所定のブロックデータサイズで分割した領域を示すブロックの単位で、バッファ部を管理するバッファ管理部をさらに備え、バッファ部には、バッファ管理部によって、ファイルを構成する複数のブロックにそれぞれ対応する複数のブロック領域が確保され、ファイル書き込み処理要求のデータが、ファイル上で対応する当該ブロック領域に格納され、判定部は、所定の判定条件に基づいて、領域の一部にのみデータが格納されたブロック領域を読み出し対象として設定し、領域の全てにデータが格納されたブロック領域を書き込み対象として設定し、遅延読み出し処理部は、読み出し対象に設定されたブロック領域に対応するファイルデータの遅延読み出し処理要求を出力し、記憶部から読み出されたファイルデータの全部又は一部をバッファ部に格納し、遅延書き込み処理部は、書き込み対象と設定されたブロック領域に対応するファイルデータの遅延書き込み処理要求を出力し、当該ブロック領域に格納されているファイルデータの全部又は一部を記憶部へ書き込むことを特徴とする。

【0018】上記のように、第3の発明によれば、ファイルデータの読み出し処理を所定のブロック領域の単位で行い、全てのデータが揃っていないブロック領域のデータだけを記憶部から読み出す。これにより、任意サイズのファイル書き込み処理要求に対しても、リアルタイムにファイルデータの書き込み処理を行うことができる。

【0019】第4の発明は、第3の発明に従属する発明であって、判定部は、最後に入力したファイル書き込み処理要求に従って格納したデータのバッファ部上の格納

終了位置を、ファイル先頭からのオフセット位置で示される最終アクセス位置として保存し、新たに入力したファイル書き込み処理要求に従って新たに格納するデータのバッファ部上のファイル先頭からのオフセット位置で示される格納開始位置が、最終アクセス位置と連続するか否かを判定し、位置が連続しないと判定した場合、最終アクセス位置を含むブロック領域の全てにデータが格納されていないければ、当該ブロック領域を読み出し対象に設定すると共に、当該ブロック領域中のデータが存在しない範囲を遅延読み出しデータ範囲として設定し、位置が連続しないと判定した場合、格納開始位置を含むブロック領域の先頭位置から格納開始位置までにデータが存在しない範囲があれば、当該ブロック領域を読み出し対象と設定すると共に、当該ブロック領域中の当該範囲を遅延読み出しデータ範囲として設定し、遅延読み出し処理部は、遅延読み出し処理要求によって読み出されたファイルデータの全部又は遅延読み出しデータ範囲に対応する部分のみを、バッファ部の該当するブロック領域又は遅延読み出しデータ範囲に格納することを特徴とする。

【0020】上記のように、第4の発明によれば、ファイル書き込み処理要求によって不連続な位置が指定された場合に限り、記憶部からの読み出し処理が行われる。従って、読み出し処理に使用する帯域幅を従来に比べ削減でき、有効利用することができる。

【0021】第5の発明は、第3及び第4の発明に従属する発明であって、判定部は、領域の一部にのみデータが格納されたが読み出し対象に設定しなかったブロック領域が、予め定められた判定時間を経過した後も書き込み対象に設定できない場合には、当該ブロック領域を読み出し対象に設定すると共に、当該ブロック領域中のデータが存在しない範囲を遅延読み出しデータ範囲として設定することを特徴とする。

【0022】上記のように、第5の発明によれば、記憶部に書き込まれる前のデータをバッファ部に長時間保持させておくことがなくなるので、システム障害時のフォールトトレランスが実現できる。

【0023】第6の発明は、第3～第5の発明に従属する発明であって、記憶部に対するアクセスは、固定長の単位に基づく所定のアクセスデータサイズで行われ、ブロックデータサイズは、アクセスデータサイズのN倍（Nは、自然数）であり、判定部は、読み出し対象に設定すべきブロック領域の、データが格納されている範囲のデータサイズが、アクセスデータサイズのn倍（nは、自然数）であり、かつ、当該範囲の先頭位置が、アクセスデータサイズでファイルが分割されたときの区切り位置に一致する場合には、当該ブロック領域を書き込み対象に設定し、遅延書き込み処理部は、ブロック領域内の末格納範囲へのデータ読み出し処理が行われることなく、範囲のデータのみを記憶部の対応する位置へ書き

込むことを特徴とする。

【0024】上記のように、第6の発明によれば、ブロック領域のサイズにかかわらず記憶部への書き込みアクセスが可能なサイズであれば、読み出し処理により不足部分を補うことなく、ファイルデータの書き込みを行う。これにより、帯域幅を無駄にせずにリアルタイムなファイルデータの書き込み処理を行うことができる。

【0025】第7の発明は、第6の発明に従属する発明であって、遅延読み出し処理部は、読み出し対象に設定されたブロック領域の、遅延読み出しデータ範囲が、アクセスデータサイズの $m$ 倍 ( $m$ は、自然数) であり、かつ、当該遅延読み出しデータ範囲の先頭位置が、アクセスデータサイズでファイルが分割されたときの区切り位置に一致する場合には、ファイルデータ内の当該遅延読み出しデータ範囲に該当する部分のみを、記憶部からバッファ部へ読み出すことを特徴とする。

【0026】上記のように、第7の発明によれば、ブロック領域のサイズにかかわらず記憶部からの読み出しアクセスが可能なサイズであれば、ブロック領域全体を読み出すことなく不足部分 (遅延読み出しデータ範囲) のみのデータを読み出す。これにより、帯域幅を無駄にせずにリアルタイムなファイルデータの読み出し処理を行うことができる。

【0027】第8の発明は、第3～第7の発明に従属する発明であって、判定部は、バッファ部にすでに格納されているファイル書き込み処理要求の対象となるファイルのデータサイズが、予め定められたバッファデータサイズを超えない場合のみ、新たに入力したファイル書き込み処理要求に従ったバッファ部へのデータの格納を行うことを特徴とする。

【0028】第9の発明は、第8の発明に従属する発明であって、バッファデータサイズは、ブロックデータサイズの $K$ 倍 ( $K$ は、自然数) であることを特徴とする。

【0029】上記のように、第8及び第9の発明によれば、各ファイルの帯域幅に応じて、バッファ部の領域を有効利用することができる。

【0030】第10の発明は、アクセス可能な固定長の単位でデータが管理される記憶部に対して、入力されるファイル操作要求に応じたファイル処理を実行するファイルシステムであって、データが一時的に格納されるバッファ部と、外部から入力される帯域設定要求によって指定される又は予め内部に有する情報から取得する帯域幅に基づいて、全ファイル又はファイル毎に予め定められた書き込み周期を設定する帯域設定部と、入力される任意サイズのデータによるファイル書き込み処理要求に従って、当該データをバッファ部へ格納すると共に、バッファ部でのファイルデータの格納状態に基づいて、ファイルデータの読み出し対象及び書き込み対象を設定する判定部と、設定された読み出し対象のファイルデータに応じた遅延読み出し処理要求を、ファイルデータの格

納位置が、ファイル内の相対的なアドレス情報で示されるファイル形式又は記憶部内の絶対的なアドレス情報で示されるデータ形式のいずれかを用いて記憶部へ出力することにより、当該読み出し対象のファイルデータを、記憶部からバッファ部へ読み出す遅延読み出し処理部と、書き込み周期に従って、設定された書き込み対象のファイルデータに応じた遅延書き込み処理要求を、ファイル形式又はデータ形式のいずれかを用いて記憶部へ出力することにより、当該書き込み対象のファイルデータを、バッファ部から記憶部へ書き込む遅延書き込み処理部とを備える。

【0031】上記のように、第10の発明によれば、記憶部へファイルデータを書き込むにあたり、バッファ部でのファイルデータの格納状態に基づいて、読み出しが必要なデータだけを記憶部から読み出す。これにより、任意サイズのファイル書き込み処理要求に対しても、リアルタイムにファイルデータの書き込み処理を行うことができる。また、常にキャッシュメモリにデータを先読みしておく従来の方式に比べて読み出し処理の無駄がなく、帯域幅の有効利用ができる。さらに、データ転送速度が異なるファイルに対しても帯域幅を無駄にせず、適切な帯域幅での書き込みが行える。

【0032】第11の発明は、第10の発明に従属する発明であって、ファイルを所定のブロックデータサイズで分割した領域を示すブロックの単位で、バッファ部を管理するバッファ管理部をさらに備え、バッファ部には、バッファ管理部によって、ファイルを構成する複数のブロックにそれぞれ対応する複数のブロック領域が確保され、ファイル書き込み処理要求のデータが、ファイル上で対応する当該ブロック領域に格納され、判定部は、所定の判定条件に基づいて、領域の一部にのみデータが格納されたブロック領域を読み出し対象として設定し、領域の全てにデータが格納されたブロック領域を書き込み対象として設定し、遅延読み出し処理部は、読み出し対象に設定されたブロック領域に対応するファイルデータの遅延読み出し処理要求を出力し、記憶部から読み出されたファイルデータの全部又は一部をバッファ部に格納し、遅延書き込み処理部は、書き込み対象と設定されたブロック領域に対応するファイルデータの遅延書き込み処理要求を出力し、当該ブロック領域に格納されているファイルデータの全部又は一部を記憶部へ書き込むことを特徴とする。

【0033】上記のように、第11の発明によれば、ファイルデータの読み出し処理を所定のブロック領域の単位で行い、全てのデータが揃っていないブロック領域のデータだけを記憶部から読み出す。これにより、任意サイズのファイル書き込み処理要求に対しても、リアルタイムにファイルデータの書き込み処理を行うことができる。

【0034】第12の発明は、第11の発明に従属する

発明であって、判定部は、最後に入力したファイル書き込み処理要求に従って格納したデータのバッファ部上の格納終了位置を、ファイル先頭からのオフセット位置で示される最終アクセス位置として保存し、新たに入力したファイル書き込み処理要求に従って新たに格納するデータのバッファ部上のファイル先頭からのオフセット位置で示される格納開始位置が、最終アクセス位置と連続するか否かを判定し、位置が連続しないと判定した場合、最終アクセス位置を含むブロック領域の全てにデータが格納されていなければ、当該ブロック領域を読み出し対象に設定すると共に、当該ブロック領域中のデータが存在しない範囲を遅延読み出しデータ範囲として設定し、位置が連続しないと判定した場合、格納開始位置を含むブロック領域の先頭位置から格納開始位置までにデータが存在しない範囲があれば、当該ブロック領域を読み出し対象と設定すると共に、当該ブロック領域中の当該範囲を遅延読み出しデータ範囲として設定し、遅延読み出し処理部は、遅延読み出し処理要求によって読み出されたファイルデータの全部又は遅延読み出しデータ範囲に対応する部分のみを、バッファ部の該当するブロック領域又は遅延読み出しデータ範囲に格納することを特徴とする。

【0035】上記のように、第12の発明によれば、ファイル書き込み処理要求によって不連続な位置が指定された場合に限り、記憶部からの読み出し処理が行われる。従って、読み出し処理に使用する帯域幅を従来に比べ削減でき、有効利用することができる。

【0036】第13の発明は、第11及び第12の発明に従属する発明であって、帯域設定部は、帯域設定要求によって指定される又は情報から取得する帯域幅に基づいて、全ファイル又はファイル毎に判定時間を設定し、判定部は、領域の一部にのみデータが格納されたが読み出し対象に設定しなかったブロック領域が、判定時間を経過した後も書き込み対象に設定できない場合には、当該ブロック領域を読み出し対象に設定すると共に、当該ブロック領域中のデータが存在しない範囲を遅延読み出しデータ範囲として設定することを特徴とする。

【0037】上記のように、第13の発明によれば、記憶部に書き込まれる前のデータをバッファ部に長時間保持させておくことがなくなるので、システム障害時のフォールトトレランスが実現できる。

【0038】第14の発明は、第11～第13の発明に従属する発明であって、記憶部に対するアクセスは、固定長の単位に基づく所定のアクセスデータサイズで行われ、ブロックデータサイズは、アクセスデータサイズのN倍であり、判定部は、読み出し対象に設定すべきブロック領域の、データが格納されている範囲のデータサイズが、アクセスデータサイズのn倍であり、かつ、当該範囲の先頭位置が、アクセスデータサイズでファイルが分割されたときの区切り位置に一致する場合には、当該

ブロック領域を書き込み対象に設定し、遅延書き込み処理部は、ブロック領域内の未格納範囲へのデータ読み出し処理が行われることなく、範囲のデータのみを記憶部の対応する位置へ書き込むことを特徴とする。

【0039】上記のように、第14の発明によれば、ブロック領域のサイズにかかわらず記憶部への書き込みアクセスが可能なサイズであれば、読み出し処理により不足部分を補うことなく、ファイルデータの書き込みを行う。これにより、帯域幅を無駄にせずにリアルタイムなファイルデータの書き込み処理を行うことができる。

【0040】第15の発明は、第14の発明に従属する発明であって、遅延読み出し処理部は、読み出し対象に設定されたブロック領域の、遅延読み出しデータ範囲が、アクセスデータサイズのm倍であり、かつ、当該遅延読み出しデータ範囲の先頭位置が、アクセスデータサイズでファイルが分割されたときの区切り位置に一致する場合には、ファイルデータ内の当該遅延読み出しデータ範囲に該当する部分のみを、記憶部からバッファ部へ読み出すことを特徴とする。

【0041】上記のように、第15の発明によれば、ブロック領域のサイズにかかわらず記憶部からの読み出しアクセスが可能なサイズであれば、ブロック領域全体を読み出すことなく不足部分（遅延読み出しデータ範囲）のみのデータを読み出す。これにより、帯域幅を無駄にせずにリアルタイムなファイルデータの読み出し処理を行うことができる。

【0042】第16の発明は、第11～第15の発明に従属する発明であって、帯域設定部は、帯域設定要求によって指定される又は情報から取得する帯域幅に基づいて、バッファデータサイズを設定し、判定部は、バッファ部にすでに格納されているファイル書き込み処理要求の対象となるファイルのデータサイズが、バッファデータサイズを超えない場合のみ、新たに入力したファイル書き込み処理要求に従ったバッファ部へのデータの格納を行うことを特徴とする。

【0043】第17の発明は、第16の発明に従属する発明であって、帯域設定部は、バッファデータサイズを、ブロックデータサイズのK倍に設定することを特徴とする。

【0044】上記のように、第16及び第17の発明によれば、各ファイルの帯域幅に応じて、バッファ部の領域を有効利用することができる。

【0045】第18の発明は、アクセス可能な固定長の単位でデータが管理される記憶部に対して、入力されるファイル操作要求に応じたファイル処理を実行するファイルシステムであって、データが一時的に格納されるバッファ部と、入力される所定の固定サイズのデータによるファイル書き込み処理要求に従って、当該データをバッファ部へ格納すると共に、バッファ部でのファイルデータの格納状態に基づいて、ファイルデータの書き込み

対象を設定する判定部と、設定された書き込み対象のファイルデータに応じた遅延書き込み処理要求を、ファイルデータの格納位置がファイル内の相対的なアドレス情報で示されるファイル形式を用いて記憶部へ出力することにより、当該書き込み対象のファイルデータを、バッファ部から記憶部へ書き込む遅延書き込み処理部とを備える。

【0046】上記のように、第18の発明によれば、バッファ部でのファイルデータの格納状態に基づいて、書き込み可能なファイルデータから順に記憶部へ書き込む。これにより、任意サイズのファイル書き込み処理要求に対しても、リアルタイムにファイルデータの書き込み処理を行うことができる。また、常にキャッシュメモリにデータを先読みしておく従来方式のように読み出し処理が不要なので、帯域幅の有効利用ができる。

【0047】第19の発明は、第18の発明に従属する発明であって、遅延書き込み処理部は、全ファイル又はファイル毎に予め定められた書き込み周期に従って、遅延書き込み処理要求を出力することを特徴とする。

【0048】上記のように、第19の発明によれば、データ転送速度が異なるファイルに対しても帯域幅を無駄にせず、適切な帯域幅での書き込みが行える。

【0049】第20の発明は、第18及び第19の発明に従属する発明であって、ファイルを所定のブロックデータサイズで分割した領域を示すブロックの単位で、バッファ部を管理するバッファ管理部をさらに備え、バッファ部には、バッファ管理部によって、ファイルを構成する複数のブロックにそれぞれ対応する複数のブロック領域が確保され、ファイル書き込み処理要求のデータが、ファイル上で対応する当該ブロック領域に格納され、判定部は、所定の判定条件に基づいて、領域の全てにデータが格納されたブロック領域を書き込み対象として設定し、遅延書き込み処理部は、書き込み対象と設定されたブロック領域に対応するファイルデータの遅延書き込み処理要求を出力し、当該ブロック領域に格納されているファイルデータの全部又は一部を記憶部へ書き込むことを特徴とする。

【0050】上記のように、第20の発明によれば、ファイルデータの書き込み処理を所定のブロック領域の単位で行い、全てのデータが揃ったブロック領域のデータから順に記憶部へ書き込む。これにより、任意サイズのファイル書き込み処理要求に対しても、リアルタイムにファイルデータの書き込み処理を行うことができる。

【0051】第21の発明は、第20の発明に従属する発明であって、判定部は、最後に入力したファイル書き込み処理要求に従って格納したデータのバッファ部上の格納終了位置を、ファイル先頭からのオフセット位置で示される最終アクセス位置として保存し、新たに入力したファイル書き込み処理要求に従って新たに格納するデータのバッファ部上のファイル先頭からのオフセット位

置で示される格納開始位置が、最終アクセス位置と連続するかどうかを判定し、位置が連続しないと判定した場合、最終アクセス位置を含むブロック領域中のデータが存在する範囲を遅延書き込みデータ範囲として設定し、遅延書き込み処理部は、遅延書き込みデータ範囲のデータのみを記憶部へ書き込むことを特徴とする。

【0052】上記のように、第21の発明によれば、ファイル書き込み処理要求によって不連続な位置が指定された場合は、遅延書き込みデータ範囲のデータのみを記憶部へ書き込む。従って、帯域幅を有効利用することができる。

【0053】第22の発明は、第21の発明に従属する発明であって、判定部は、バッファ部にすでに格納されているファイル書き込み処理要求の対象となるファイルのデータサイズが、予め定められたバッファデータサイズを超えない場合のみ、新たに入力したファイル書き込み処理要求に従ったバッファ部へのデータの格納を行うことを特徴とする。

【0054】第23の発明は、第22の発明に従属する発明であって、バッファデータサイズは、ブロックデータサイズのK倍であることを特徴とする。

【0055】上記のように、第22及び第23の発明によれば、各ファイルの帯域幅に応じて、バッファ部の領域を有効利用することができる。

【0056】第24の発明は、アクセス可能な固定長の単位でデータが管理される記憶部に対して、入力されるファイル操作要求に応じたファイル処理を実行するファイルシステムであって、データが一時的に格納されるバッファ部と、外部から入力される帯域設定要求によって指定される又は予め内部に有する情報から取得する帯域幅に基づいて、全ファイル又はファイル毎に予め定められた書き込み周期を設定する帯域設定部と、入力される所定の固定サイズのデータによるファイル書き込み処理要求に従って、当該データをバッファ部へ格納すると共に、バッファ部でのファイルデータの格納状態に基づいて、ファイルデータの書き込み対象を設定する判定部と、書き込み周期に従って、設定された書き込み対象のファイルデータに応じた遅延書き込み処理要求を、ファイルデータの格納位置がファイル内の相対的なアドレス情報で示されるファイル形式を用いて記憶部へ出力することにより、当該書き込み対象のファイルデータを、バッファ部から記憶部へ書き込む遅延書き込み処理部とを備える。

【0057】上記のように、第24の発明によれば、バッファ部でのファイルデータの格納状態に基づいて、書き込み可能なファイルデータから順に記憶部へ書き込む。これにより、任意サイズのファイル書き込み処理要求に対しても、リアルタイムにファイルデータの書き込み処理を行うことができる。また、常にキャッシュメモリにデータを先読みしておく従来方式のように読み出し

処理が不要なので、帯域幅の有効利用ができる。さらに、データ転送速度が異なるファイルに対しても帯域幅を無駄にせず、適切な帯域幅での書き込みが行える。

【0058】第25の発明は、第24の発明に従属する発明であって、ファイルを所定のブロックデータサイズで分割した領域を示すブロックの単位で、バッファ部を管理するバッファ管理部をさらに備え、バッファ部には、バッファ管理部によって、ファイルを構成する複数のブロックにそれぞれ対応する複数のブロック領域が確保され、ファイル書き込み処理要求のデータが、ファイル上で対応する当該ブロック領域に格納され、判定部は、所定の判定条件に基づいて、領域の全てにデータが格納されたブロック領域を書き込み対象として設定し、遅延書き込み処理部は、書き込み対象と設定されたブロック領域に対応するファイルデータの遅延書き込み処理要求を出力し、当該ブロック領域に格納されているファイルデータの全部又は一部を記憶部へ書き込むことを特徴とする。

【0059】上記のように、第25の発明によれば、ファイルデータの書き込み処理を所定のブロック領域の単位で行い、全てのデータが揃ったブロック領域のデータから順に記憶部へ書き込む。これにより、任意サイズのファイル書き込み処理要求に対しても、リアルタイムにファイルデータの書き込み処理を行うことができる。

【0060】第26の発明は、第25の発明に従属する発明であって、判定部は、最後に入力したファイル書き込み処理要求に従って格納したデータのバッファ部上の格納終了位置を、ファイル先頭からのオフセット位置で示される最終アクセス位置として保存し、新たに入力したファイル書き込み処理要求に従って新たに格納するデータのバッファ部上のファイル先頭からのオフセット位置で示される格納開始位置が、最終アクセス位置と連続するか否かを判定し、位置が連続しないと判定した場合、最終アクセス位置を含むブロック領域中のデータが存在する範囲を遅延書き込みデータ範囲として設定し、遅延書き込み処理部は、遅延書き込みデータ範囲のデータのみを記憶部へ書き込むことを特徴とする。

【0061】上記のように、第26の発明によれば、ファイル書き込み処理要求によって不連続な位置が指定された場合は、遅延書き込みデータ範囲のデータのみを記憶部へ書き込む。従って、帯域幅を有効利用することができる。

【0062】第27の発明は、第25及び第26の発明に従属する発明であって、帯域設定部は、帯域設定要求によって指定される又は情報から取得する帯域幅に基づいて、バッファデータサイズを設定し、判定部は、バッファ部にすでに格納されているファイル書き込み処理要求の対象となるファイルのデータサイズが、バッファデータサイズを超えない場合のみ、新たに入力したファイル書き込み処理要求に従ったバッファ部へのデータの格

納を行うことを特徴とする。

【0063】第28の発明は、第27の発明に従属する発明であって、帯域設定部は、バッファデータサイズを、ブロックデータサイズのK倍に設定することを特徴とする。

【0064】上記のように、第27及び第28の発明によれば、各ファイルの帯域幅に応じて、バッファ部の領域を有効利用することができる。

【0065】第29の発明は、入力されるファイル操作の要求に応じたファイル処理を記憶部に対して実行するファイルサーバであって、請求項1～28のいずれかに記載のファイルシステムと、アクセス可能な固定長の単位でデータが管理される記憶部とを備え、ファイルシステムは、ファイル処理の要求を、ファイルデータの格納位置が記憶部内の絶対的なアドレス情報で示されるデータ形式を用いて記憶部へ出力することを特徴とする。

【0066】第30の発明は、ファイル操作に関するファイル処理の要求を出力する端末と、当該要求に応じた処理を実行するファイルサーバとが、電気通信回線により接続されて構成されるネットワークシステムであって、端末は、請求項1～28のいずれかに記載のファイルシステムを備え、ファイル処理の要求を、ファイルデータの格納位置がファイル内の相対的なアドレス情報で示されるファイル形式を用いてファイルサーバへ出力し、ファイルサーバは、アクセス可能な固定長の単位でデータが管理される記憶部を備え、電気通信回線を介して端末からファイル処理の要求を入力し、当該要求に応じた処理を当該記憶部に対して行うことを特徴とする。

【0067】

【発明の実施の形態】以下、本発明の実施の形態について、図面を参照しながら説明する。

(ファイルシステムの適用環境) 図1は、本発明によって提供されるファイルシステムが適用されるクライアント・サーバ・システムの構成例を示すブロック図である。図1において、クライアント・サーバ・システムは、複数の端末(クライアント)11～1N(Nは、2以上の整数)と、ファイルサーバ20とが、ネットワーク30を介して互いに接続されている構成である。これら複数の端末11～1N及びファイルサーバ20は、コンピュータ装置によって実現される。一方、ネットワーク30は、例えば、ATMやイーサネット(登録商標)等の有線ネットワークや、W-CDMA等の無線ネットワークによって実現される。

【0068】(第1の実施形態) 本発明の第1の実施形態に係るファイルシステムは、上記クライアント・サーバ・システムのファイルサーバ20に適用されるリアルタイム・ローカル・ファイルシステムである。図2は、本発明の第1の実施形態に係るファイルシステム22を用いた図1のファイルサーバ20の詳細な構成を示すブロック図である。図2において、ファイルサーバ20

は、サーバアプリケーション 21 と、ファイルシステム 22 と、ディスクアクセス変換部 23 と、データ記憶部 24 と、プロトコル処理部 25 とを備えている。まず、図 2 に示すファイルサーバ 20 の各構成を説明する。

【0069】サーバアプリケーション 21 は、ファイルサーバ 20 の設定・管理を行うものであり、端末 11 ~ 1N から与えられる要求を受け付け、ファイルシステム 22 へファイルの読み出し要求及び書き込み要求を伝える。ファイルシステム 22 は、データをファイルとして扱えるようにデータ記憶部 24 に格納する機能と、格納されているファイルに対するオープン、読み出し、書き込み及びクローズ等の機能とを備えるものである。なお、本発明のファイルシステムで取り扱われるファイルは、主にビデオやオーディオ等の高速かつリアルタイムなデータ転送が必要とされるファイルとする。また、上述のファイルオープンやファイル書き込み等のファイルアクセスは、通常の文字情報や静止画像を対象としたファイルシステムと同様に、汎用のインタフェースを使ってアプリケーションから要求されるものとする。特に、ファイルの読み出し及び書き込みについては、任意のデータサイズで要求できるものとする。ディスクアクセス変換部 23 は、ファイルシステム 22 から与えられるファイルの読み出し要求及び書き込み要求を、データ記憶部 24 で識別できるコマンドに変換して発行する。例えば、SCSI のハードディスクにコマンドを発行する SCSI ディスクドライバ等が、ディスクアクセス変換部 23 に相当する。データ記憶部 24 は、データを記憶する装置であり、ハードディスク (HD) や光磁気ディスク等のランダムアクセス可能なディスク型の記録媒体を駆動できる記憶装置である。プロトコル処理部 25 は、ファイルサーバ 20 と端末 11 ~ 1N との間のデータ転送に必要なプロトコル変換を行うものである。

【0070】以上のような構成により、ファイルサーバ 20 は、端末 11 ~ 1N からファイルの書き込み要求があったデータをファイルシステム 22 を介してデータ記憶部 24 へ格納する、また、ファイルの読み出し要求があったデータを、ファイルシステム 22 を介してデータ記憶部 24 から端末 11 ~ 1N へ送出する、等の様々なファイル処理を行う。本発明が提供するファイルシステムは、これらのファイル処理の中でファイル書き込み処理に特徴があるため、以下の各実施形態においては、ファイル書き込み処理について説明する。

【0071】なお、データ記憶部 24 には、データ読み出し／書き込みの要求を受け付けることのできる最小サイズ (セクタサイズ = 512 バイト等) が予め定められており、以後このサイズを「最小アクセス単位」と呼ぶことにする。同様に、ディスクアクセス変換部 23 にも、ファイルシステム 22 からアクセス可能な最小サイズが予め定められているものとする。以後このサイズを「アクセス単位」と呼ぶことにする。このアクセス単位

は、最小アクセス単位と同一サイズでも倍数のサイズでもよい。さらに、通常、ファイルシステム 22 がディスクアクセス変換部 23 へアクセスする単位を、最小アクセス単位及びアクセス単位とは別に「ブロック」と呼ぶことにする (なお、ブロックのサイズを「BLOCK SIZE」と表記する)。これは、一般に、記憶装置に対して小さな単位で読み書きするほど転送サイズ当たりの処理時間が長くなることを考慮し、最小アクセス単位より大きな「ブロック」というサイズで、記憶装置にアクセスするようにしているからである。なお、これらの単位の関係を図 3 に示しておく。

【0072】次に、本発明の第 1 の実施形態に係るファイルシステム 22 の各構成を説明する。図 2 において、第 1 の実施形態に係るファイルシステム 22 は、ブロック判定部 221 と、バッファ管理部 222 と、遅延書き込み用バッファ 223 と、遅延書き込み処理部 224 と、遅延読み出し処理部 225 とを備えている。なお、図 2 中、太線矢印はデータの流れを示し、細線矢印は要求又は指示の流れを示している。

【0073】バッファ管理部 222 は、サーバアプリケーション 21 からファイルオープンの要求を受け付けた場合には、メモリ上に必要な遅延書き込み用バッファ 223 の領域を確保する。ここで、バッファ管理部 222 は、要求内容に応じて、複数ブロック分の領域を遅延書き込み用バッファ 223 として確保する。この確保された領域に関する情報は、ブロック判定部 221 に伝えられる。なお、この情報は、直接ではなく遅延書き込み用バッファ 223 を介してブロック判定部 221 に伝えられてもよい。一方、バッファ管理部 222 は、サーバアプリケーション 21 からファイルクローズの要求を受け付けた場合には、確保した遅延書き込み用バッファ 223 の領域を解放する。なお、遅延書き込み用バッファ 223 の領域の確保及び開放は、上述したファイルオープン及びファイルクローズ時に行われる方法以外にも、次のような方法で実現可能である。まず、システム起動時 (PC 電源投入時) に、バッファ管理部 222 が、大量の遅延書き込み用バッファ 223 の領域をアロケートしておき、フリーバッファ (割り当てられていない領域) として管理しておく。そして、ファイルオープン時には、フリーバッファの中から必要なブロック数分だけが確保される (割り当てられる)。一方、ファイルクローズ時には、割り当てられていた領域が実際に開放されるのではなく、バッファ管理部 222 で管理されるフリーバッファとして返却される。

【0074】ブロック判定部 221 は、サーバアプリケーション 21 からファイルの書き込み要求を受け付けた場合、要求されたデータを、バッファ管理部 222 によって確保された遅延書き込み用バッファ 223 へ転送する。さらに、ブロック判定部 221 は、遅延書き込み用バッファ 223 に格納されているデータと、サーバアプ

リケーション21から転送されたデータとに基づいて、ブロック単位のデータが揃ったかどうかを判定し、必要に応じて遅延書き込み処理部224及び遅延読み出し処理部225へ特定の処理を依頼する。

【0075】ここで、図4を用いて、ブロック判定部221で行われる判定方法の概要を説明する。図4において、ブロック[0]～ブロック[4]は、遅延書き込み用バッファ223に確保された領域であって、ファイル先頭位置からのオフセットをブロック単位で区切ったものである。例えば、ブロック[0]は、ファイル先頭位置からBLKSIZE分のデータ範囲を意味する。また、図中のWriteFile1～WriteFile6は、サーバアプリケーション21から与えられる1回目～6回目の書き込み要求を表し、その矢印は、要求によって指定されたデータ範囲を示す。

【0076】まず、WriteFile1のデータ範囲は、ブロック[0]と同じなので、書き込むべきデータが遅延書き込み用バッファ223内に1ブロック分揃っている。従って、ブロック判定部221は、ブロック[0]のデータがデータ記憶部24に書き込み可能であると判定する。次に、WriteFile2を受け付けた場合には、データ範囲がブロック[1]の一部分なので、ブロック判定部221は、ブロック[1]のデータが書き込みできない状態であると判定する。ただし、その後を受け付ける書き込み要求によって、ブロック

[1]のデータが揃う可能性があるので、ブロック判定部221は、ブロック[1]をそのまま保留状態とする。そして、次にWriteFile3及びWriteFile4を受け付けた場合に、初めてブロック[1]のデータが揃うので、ブロック判定部221は、ブロック[1]のデータが書き込み可能であると判定する。一方、ブロック[2]のようにブロックの一部分しかデータが揃っていないときに、WriteFile5のようにデータ範囲が不連続に要求されてしまった場合には、今後、その空き部分を埋めるような要求がサーバアプリケーション21から発行される可能性が低い。従って、ブロック判定部221は、ブロック[2]及びブロック[3]におけるデータの無い範囲（遅延書き込み用バッファ223上にデータが格納されていない範囲）を「データ無効範囲」とみなし、この範囲をデータ記憶部24から読み出したデータで埋めるように、遅延読み出し処理部225へ指示する。

【0077】このように、本発明では、不連続な書き込み要求を受けてデータ無効範囲が発生した場合にだけ、遅延読み出し処理部225によるデータ読み出し処理が行われる。従って、WriteFile2やWriteFile4を受け付けた場合に、ブロック[1]やブロック[2]のデータをキャッシュとして読み出していた従来技術に比べ、本発明では、後からの書き込み要求でデータの揃う可能性のあるブロックをそのまま保留して

おき、直ぐにキャッシュ読み出しを行わないようにしている。

【0078】遅延書き込み処理部224は、一定周期毎に遅延書き込み用バッファ223の状態を調べ、データ記憶部24へ書き込み可能な状態にあるブロックのデータのライト要求(Write)を、ディスクアクセス変換部23へ発行する。遅延読み出し処理部225は、ブロック判定部221からデータ無効範囲のデータ読み出しを指示された場合、該当するデータのリード要求(Read)をディスクアクセス変換部23へ発行する。

【0079】次に、上記のように構成されたファイルシステム22が行う動作の内、ファイルオープン処理、ファイル書き込み処理及びファイルクローズ処理の動作を、図5～図13を用いて順に説明する。図5及び図6は、ファイルシステム22がサーバアプリケーション21から書き込み要求を受け付けた場合、ブロック判定部221が行う動作を示すフローチャートである。図7は、遅延書き込み処理部224が行う動作を示すフローチャートである。図8は、遅延読み出し処理部225が行う動作を示すフローチャートである。図9は、データ記憶部24から読み出されたデータが遅延書き込み用バッファ223へデータ転送される方法の一例を示す図である。図10～図13は、書き込み要求のデータ範囲とブロックの区切り位置との関係の一例を示す図である。なお、以下の説明では、端末11から書き込み要求を受け付けた場合を一例に挙げて、各動作の説明を行う。

【0080】(1) ファイルオープン処理

端末11から、ファイルサーバ20が有するファイルについてオープン処理が要求されると、この要求は、プロトコル処理部25でプロトコル変換された後、サーバアプリケーション21に伝えられる。サーバアプリケーション21は、この要求をファイルオープン要求としてファイルシステム22へ発行する。ファイルシステム22は、ファイルオープン要求を受けると、要求されたファイルの有無や状態を判断して、オープン処理の成功/失敗を決定する。具体的には、ファイルシステム22は、要求されたファイルがデータ記憶部24に存在し、かつ、オープン可能な状態であれば、オープン処理を成功とする。一方、ファイルシステム22は、要求されたファイルがデータ記憶部24に無かったり、他の端末によって排他的にオープンされている等の理由でオープン不可能な状態であれば、オープン処理を失敗とする。オープン処理が成功の場合、ファイルシステム22は、バッファ管理部222によってメモリ内に遅延書き込み用バッファ223の領域を確保すると共に、オープン処理がなされたファイルの識別子をサーバアプリケーション21に返送する。サーバアプリケーション21が受け取ったファイル識別子は、プロトコル処理部25を介して端末11に送られる。これにより、以後の書き込み等のファイルアクセスが、可能な状態となる。



**【0081】 (2) ファイル書き込み処理**

次に、端末11が、上記(1)ファイルオープン処理で得たファイル識別子を用いて、ファイル書き込みを行う動作を説明する。なお、ここでは既存ファイルについて変更や追加を行い、再度ファイルを書き込む例を説明するが、ファイルを新規に作成する場合の書き込みについても、同様の動作で可能である。端末11は、ファイルサーバ20へ、ファイル識別子を用いて書き込み要求を送出し、続けて書き込みたいデータを送出する。この書き込み要求は、プロトコル処理部25によってプロトコル処理された後に、サーバアプリケーション21へ伝えられ、次いでファイルシステム22へ伝えられる。

**【0082】**(書き込み要求受付処理) ファイルシステム22が書き込み要求を受け取ると、ブロック判定部221によって図5に示す書き込み要求受付処理が実行される。まず、今回受け付けた書き込み要求のデータ範囲、すなわちデータ開始位置(以下、StartPntと記す)からデータ終了位置(以下、EndPntと記す)が、前回受け付けた書き込み要求のデータと連続しているか否かが判定される(ステップS501)。このステップS501では、記憶している前回の書き込み要求のデータ終了位置(以下、PreEndPntと記す)が今回の書き込み要求のStartPntと一致する場合に、連続アクセスであると判定される。なお、StartPnt及びEndPntは、共にファイル先頭からのオフセットバイト数とする。また、初期状態(ファイルオープン直後で、書き込み要求を受け取っていない状態)では、PreEndPnt=0(ファイル先頭位置)とする。ステップS501で連続アクセスであると判定された場合には、ステップS503の処理に移行する。一方、ステップS501で不連続アクセスであると判定された場合には、データ無効範囲が発生した可能性があるため、後述する不連続アクセスの読み出し処理(ステップS502、図6)が行われた後、ステップS503へ移行する。

**【0083】**上記ステップS501及びS502において判定及び必要な処理が行われると、前回の書き込み要求処理において保留状態とされたブロック、すなわち一部分のみにデータが格納されたブロックが存在するか否かが判定される(ステップS503)。ステップS503で保留状態のブロックがあると判定された場合、当該ブロックのデータが格納されていない残り部分へ、今回の書き込み要求のデータの全部又は一部が転送される(ステップS506)。一方、ステップS503で保留状態のブロックが無いと判定された場合、遅延書き込み用バッファ223として1ブロック分の領域が新たに確保され(ステップS504)、そのブロックが保留状態に初期設定される(ステップS505)。そして、新たに確保されたブロックへ、今回の書き込み要求のデータの全部又は一部が転送される(ステップS506)。こ

こで、ステップS506で行われるデータ転送は、保留状態のブロック範囲内で行われるので、次に、今回の書き込み要求のデータが全て転送されたか(すなわち、EndPntまでデータ転送が完了したか)が判断される(ステップS507)。ステップS507で全てのデータ転送が完了していないと判断された場合、ステップS506でデータ転送されたブロックを確定状態に設定した(ステップS508)後、上記ステップS504～S506が繰り返し行われる。例えば、今回の書き込み要求のデータサイズがBLKSIZE×2である場合には、ステップS506が2回行われることになる。

**【0084】**そして、ステップS507で全てのデータ転送が完了したと判断された場合、今回の書き込み要求のEndPntがブロックの区切り位置かが判定される(ステップS509)。この判定は、EndPntがBLKSIZEの倍数か否かで行われる。ステップS509でブロックの区切り位置であると判定された場合には、最後にデータ転送処理を行ったブロックに全てデータが格納されたことになるので、当該ブロックを確定状態に設定した(ステップS510)後、今回の書き込み要求のEndPntが前回の書き込み要求のデータ終了位置、すなわちPreEndPntとして保存される(ステップS511)。一方、ステップS509でブロックの区切り位置でないと判定された場合には、最後にデータ転送処理を行ったブロックにデータ格納されていない部分が残っていることになるので、このブロックを保留状態のまま、上記ステップS511が実行される。これにより、書き込み要求受付処理が完了する。

**【0085】**(不連続アクセスの読み出し処理) 次に、今回の書き込み要求が、前回の書き込み要求とは不連続な位置(不連続アクセス)である場合の処理(図5のステップS502)を、図6を用いて説明する。サーバアプリケーション21から書き込み要求を受け付け、図5のステップS501で不連続アクセスであると判定された場合には、図6に示す不連続アクセスの読み出し処理が実行される。ステップS601～S603では、前回の書き込み要求のときに保留状態となり、データ記憶部24へ書き込まれないまま遅延書き込み用バッファ223に残っているブロックの処理が行われる。ステップS604～S606では、今回の書き込み要求によって、データ無効範囲が発生したブロックの処理が行われる。

**【0086】**まず、前回の書き込み要求のPreEndPntが、ブロックの区切り位置か否かが判定される(ステップS601)。この判定は、図5のステップS509での説明と同様に、PreEndPntがBLKSIZEの倍数であるか否かで行われる。ステップS601で区切り位置であると判定された場合には、保留状態のブロックが無いと判断され、ステップS604へ処理が移行する。一方、ステップS601で区切り位置でないと判定された場合には、前回の書き込み要求での最

終ブロックが保留状態であるので、最終ブロック内のデータ無効範囲に該当するデータを読み出して最終ブロックを確定状態にした後、データ記憶部 24 へ書き込む必要がある。このため、前回の書き込み要求での最終ブロック (PreEndPnt が存在するブロック) と、今回の書き込み要求での先頭ブロック (StartPnt が存在するブロック) とが、同じか否かが判定される

(ステップ S602)。この判定を行うのは、同じブロックと判定された場合には、次のステップ S603 でデータ無効範囲の読み出し処理を行わなくても、先頭ブロックに対する読み出し処理 (後述のステップ S604 の処理) によって、まとめて処理を行えるためである。従って、ステップ S602 で同じブロックと判定された場合には、ステップ S604 へ処理が移行し、同じブロックでないと判定された場合には、前回の書き込み要求での最終ブロックのデータ (後半端数) 読み出しが、遅延読み出し処理部 225 へ要求された後 (ステップ S603)、ステップ S605 へ処理が移行する。

【0087】次に、今回の書き込み要求の StartPnt が、ブロックの区切り位置か否かが判定される (ステップ S604)。ステップ S604 で区切り位置と判定された場合には、今回の書き込み要求の先頭ブロックは、ブロックの先頭からデータが揃うことになり、StartPnt より前の部分でデータ無効範囲が発生しないので、この不連続アクセスの読み出し処理が完了する。一方、ステップ S604 で区切り位置と判定された場合には、今回の書き込み要求の先頭ブロック内の StartPnt より前の部分が、データ無効範囲となってしまう。このため、今回の書き込み要求での先頭ブロックのデータ (前半端数) 読み出しが、遅延読み出し処理部 225 へ要求された後 (ステップ S605)、ステップ S606 へ処理が移行する。そして、ステップ S603 及び S605 にて要求された遅延読み出し処理 (図 7) が実行された後 (ステップ S606)、この不連続アクセスの読み出し処理が完了し、図 5 のステップ S503 へ処理が戻る。

【0088】(遅延読み出し処理) 次に、図 2 の遅延読み出し処理部 225 における動作を、図 7 を用いて説明する。遅延読み出し処理部 225 では、ブロック判定部 221 から発行された読み出し要求 (図 6 のステップ S603 及び S605) を受けて、データ記憶部 24 から対応するデータを読み出す処理 (遅延読み出し処理) が行われる。図 7 に示す遅延読み出し処理は、サーバアプリケーション 21 からファイルオープン処理の要求があった場合に開始される。この処理が開始されると、何らかの要求が受け付けられるまで待機している (ステップ S701)。要求が受け付けられると、当該要求が遅延読み出し終了要求か否かが判断される (ステップ S702)。この遅延読み出し終了要求は、サーバアプリケーション 21 からファイルクローズ処理の要求があったと

き発行されるものであり、この要求が受け付けられた場合には、この遅延読み出し処理が終了する。一方、ステップ S702 で受け付けられた要求が遅延読み出し終了要求ではないと判断された場合には、ブロック判定部 221 からの読み出し要求であると判断され、その要求内容に基づいたリード要求がディスクアクセス変換部 23 へ発行される (ステップ S703)。

【0089】ここで、ブロック判定部 221 では、読み出し要求として少なくとも以下の内容を遅延読み出し処理部 225 へ通知するものとする。

- ・読み出すべきブロックのデータ格納位置 (ファイルオフセット位置又はデータ記憶部 24 上の論理アドレス等)

- ・読み出したデータを格納する遅延書き込み用バッファ 223 のアドレス

- ・ブロック内のデータ無効範囲 (読み出したデータを転送する範囲) を特定する情報

【0090】次に、リード要求によってディスクアクセス変換部 23 を介してデータ記憶部 24 から読み出されたブロックの内、データ無効範囲の部分のデータが、遅延書き込み用バッファ 223 の対応する位置へデータ転送される (ステップ S704)。このデータ転送の方法としては、図 9 に示すように、ディスクアクセス変換部 23 内に予めテンポラリバッファを用意しておき、データ記憶部 24 から読み出されたブロックを一旦テンポラリバッファに格納させて、その後、データ無効範囲の部分のデータだけをテンポラリバッファから遅延書き込み用バッファ 223 へコピーさせる方法がある。また、データ無効範囲がディスクアクセス変換部 23 の最小アクセス単位の倍数であれば、ディスクアクセス変換部 23 が遅延書き込み用バッファ 223 のデータ無効範囲の部分へデータを直接読み出すようにしてもよい。

【0091】次に、受け取った読み出し要求が、後半端数読み出し要求 (図 6、ステップ S603 で発行) か、前半端数読み出し要求 (図 6、ステップ S605 で発行) かが判定される (ステップ S705)。そして、ステップ S705 で判定された要求が後半端数読み出し要求である場合、読み出し先のブロックが確定状態に設定され (ステップ S706)、前半端数読み出し要求である場合、読み出し先のブロックが保留状態に設定される (ステップ S707)。これは、ステップ S603 で読み出し要求された最終ブロックには、読み出し処理が完了した後、そのままブロック判定部 221 を経由せず、遅延書き込み処理部 224 によって書き込み処理が行われるのに対し、ステップ S605 で読み出し要求された先頭ブロックには、ブロック判定部 221 が読み出し処理の完了を待ち、読み出された先頭ブロックに対して続けて処理が行われるためであり、双方のブロック状態の設定方法が異なるからである。このステップ S706 又は S707 での設定が終了すると、1 ブロック分 (読み

出し要求1個分の)読み出し処理が完了する(ステップS708)。このとき、図6のステップS606において読み出し処理完了を待つブロック判定部221へ、遅延読み出し処理が完了したことが通知される。その後、ステップS701へ処理が戻り、次の要求を受け付ける状態に移行する。

【0092】なお、この例では、遅延読み出し処理の開始及び終了が、ファイルオープン及びファイルクローズのタイミングであると説明したが、ファイル単位に行わずにファイルシステム22の起動から終了まで常に遅延読み出し処理が継続していてもよい。その場合、図7のステップS701では、オープン中の全ファイルの要求が受け付けられるものとし、ステップS703～S708では、どのファイルに対する要求かが判別され、ファイルに応じた処理が行われるものとする。

【0093】(遅延書き込み処理)次に、図2の遅延書き込み処理部224における動作を、図8を使って説明する。遅延書き込み処理部224では、上述した書き込み要求受付処理(図5)において「確定状態」と設定されたブロックのデータを、データ記憶部24に書き込む処理(遅延書き込み処理)が行われる。図8に示す遅延書き込み処理は、サーバアプリケーション21からファイルオープン処理の要求があった場合に開始される。処理が開始されると、一定時間の経過後(ステップS801)、遅延書き込み処理の終了要求が受け付けられたか否かが判定される(ステップS802)。この遅延書き込み終了要求は、サーバアプリケーション21からファイルクローズ処理の要求があったとき発行されるものであり、この要求を受け付けた場合、この遅延書き込み処理が終了する。一方、ステップS802で遅延書き込み終了要求を受け付けていないと判定された場合、確認状態のブロックがあるか否かが判断される(ステップS803)。ステップS803で確認状態のブロックがあると判断された場合、ライト要求がディスクアクセス変換部23へ発行され、対応するブロックの遅延書き込み用バッファ223上に格納されたデータが、データ記憶部24へ書き込まれる(ステップS804)。一方、ステップS803で確認状態のブロックが無いと判断された場合、及びステップS804の書き込み処理が完了した後、ステップS801へ処理が戻り、同様の処理が繰り返される。

【0094】なお、この例では、遅延書き込み処理の開始及び終了が、ファイルオープン及びファイルクローズのタイミングであると説明したが、ファイル単位に行わずにファイルシステム22の起動から終了まで常に遅延書き込み処理が継続していてもよい。その場合、図8のステップS803及びS804では、オープン中の全ファイルについて確定状態のブロックが検索され、同様の処理が行われるものとする。また、上記実施例では、遅延書き込み処理部224からデータ記憶部24への書き

込みを行う一定周期を、ステップS801において一定時間待つことにより実現しているが、タイムスロット管理によるスケジューリングを用いて実現してもよい。

【0095】このように、遅延書き込み処理部224では、書き込める状態になった(全範囲にデータが格納された)ブロックだけを、一定周期毎にデータ記憶部24へ書き込むようにしている。また、サーバアプリケーション21からの書き込み要求が、連続アクセスの繰り返しである場合には、図5のステップS502が実行されず、遅延書き込み処理部224だけの処理になるので、データ記憶部24からのデータ読み出しが全く発生しないで済む。

【0096】以上説明した本発明の第1の実施形態に係るファイルシステム22の動作を、図10～図13に示す書き込み要求を一例に挙げて確認する。ここで、上述したように、各図において、遅延書き込み用バッファ223内の全データが揃っている状態のブロックを「確定」で、一部のデータしか揃っていない状態のブロックを「保留」で、データが全く揃っていない状態のブロックを「データ無し」で、それぞれ表現している。また、ブロック判定部221が遅延読み出し処理部225に読み出し要求を発行してから、遅延読み出し処理部225の処理が完了するまでの間の状態を「読み出し待ち」で表現している。

【0097】<ケース1：連続アクセスの場合>図10は、サーバアプリケーション21から連続したデータ範囲の書き込み要求が行われる(連続アクセス)ケースを示す。まず、図10(a)は、ファイルオープン処理後の全く書き込み要求を受け付けていないときのブロック状態を示す。このとき、 $PreEndPnt=0$ 、全てのブロックはデータ無し状態である。図10(b)に、1回目の書き込み要求を受け付け、ステップS501～S511(図5)の処理を完了した後の各ブロックの状態を示す。同図では、 $StartPnt=PreEndPnt=0$ であるので、要求が連続アクセスであると判定される(ステップS501)。また、 $EndPnt$ がブロックの区切り位置ではないので(ステップS509)、ブロック[0]が保留状態に設定される。次に、図10(c)に、2回目の書き込み要求を受け付け、書き込み要求受付処理までの処理を完了した後の各ブロックの状態を示す。同図では、先頭ブロックの番号=0及び最終ブロックの番号=2なので、ブロック[2]が保留状態に、ブロック[0]及びブロック[1]が確定状態に設定される。仮に、2回目の書き込み要求の $EndPnt$ が、 $EndPnt=BLKSIZE \times 3$ (ブロック[2]の最終位置)の場合には、ステップS510において最終ブロックであるブロック[2]が確定状態に設定される。このように、ブロック判定部221では、1回以上の書き込み要求のデータがまとめられ、ブロック内のデータが全て揃ったブロックが確定状態に、一部

のデータしか揃っていないブロックが保留状態に設定される。そして、遅延書き込み処理部224では、ブロック判定部221で確定状態に設定されたブロックのみ、データ記憶部24への書き込み処理が実行される。

【0098】<ケース2：不連続アクセスの場合（PreEndPntがブロックの区切り位置以外）>図11は、サーバアプリケーション21から不連続なデータ範囲の書き込み要求が行われ（不連続アクセス）、かつ、前回の書き込み要求で保留状態のブロックが存在するケースを示す（同図（a）及び（b））。図11（b）に示すように、不連続な書き込み要求が受け付けられた場合、ステップS502（図6）の処理が実行される。この場合、前回の書き込み要求の最終ブロック（ブロック[n]）が保留状態であり、かつ、前回の書き込み要求の最終ブロックと今回の書き込み要求の先頭ブロック（ブロック[n+2]）とが異なる。このため、ステップS603においてブロック[n]の読み出し要求が発行され、ブロック[n]は、読み出し待ち状態となる。このとき、データ無効範囲は、図11（b）に示す範囲となる。その後、StartPntがブロックの区切り位置なので、そのままステップS502の処理が完了し、以後ステップS503～S511の処理が行われる。なお、ブロック判定部221では、ステップS603の処理で発行された読み出し要求の処理完了を待たずに、ステップS503以後の処理が行われるため、図11（c）のように、書き込み要求処理が完了した時点でも、ブロック[n]が読み出し待ち状態のままの場合もある。ブロック[n]は、遅延読み出し処理部225によって確定状態に設定された時点で（ステップS707）、図11（d）の状態になる。

【0099】<ケース3：不連続アクセスの場合（StartPntがブロックの区切り位置以外）>図12は、不連続アクセスで、かつ、今回の書き込み要求で、ブロックの区切り位置以外の位置からデータ範囲の書き込みが行われるケースを示す（同図（a）及び（b））。StartPntが図12（b）に示す位置の場合、ブロック[n+2]の先頭からStartPntまでの範囲がデータ無効範囲となる。この場合、ステップS605においてブロック[n+2]の読み出し要求が発行され、ブロック[n+2]は読み出し待ち状態になる。読み出し要求を受けた遅延読み出し処理部225では、図7で説明した遅延読み出し処理が行われ、ステップS706においてブロック[n+2]が保留状態に設定される（図12（c））。ステップS708で読み出し処理の完了が通知されると、ブロック判定部221の読み出し処理完了待ち（ステップS606）が終了し、ステップS502の処理が終了する。その後、ブロック判定部221では、ブロック[n+2]の読み出し済みデータの後に、今回の書き込み要求のデータが転送される（ステップS506）。以降の処理は、上述した

通りであり、書き込み要求処理が完了した時点で、各ブロックが図12（d）の状態になる。

【0100】<ケース4：不連続アクセスの場合（PreEndPntとStartPntとが同じブロック内に位置する）>図13は、不連続アクセスで、かつ、前回の書き込み要求での最終ブロックと、今回の書き込み要求での先頭ブロックとが同じであるケースを示す（同図（a）及び（b））。図13（b）に示すように、PreEndPntとStartPntとが同じブロック内に位置する場合は、データ無効範囲がPreEndPntとStartPntとの間の範囲となる。従って、ステップS602からそのままステップS604に進み、ステップS605においてブロック[n]の読み出し要求が発行される。その後の動作は、図12（c）及び（d）と同じである。

#### 【0101】（3）ファイルクローズ処理

最後に、端末11がファイルの書き込みを終了し、ファイルのクローズを要求したときのクローズ処理の動作を説明する。端末11からファイルサーバ20へ、ファイルオープン処理で得られたファイル識別子を用いてファイルのクローズ処理が要求されると、この要求は、プロトコル処理部25でプロトコル変換された後、サーバアプリケーション21に伝えられる。サーバアプリケーション21は、この要求をファイルクローズ要求としてファイルシステム22へ発行する。ファイルシステム22は、ファイルクローズ要求を受けると、対応するファイルに関して、遅延書き込み用バッファ223上に保留状態のブロックがあるか否かを調べる。保留状態のブロックがあれば、ファイルシステム22は、PreEndPntから最終ブロックの最後の位置までの範囲をデータ無効範囲とみなし、この範囲のデータ読み出し及びデータ記憶部24へのデータ書き込みを行う。具体的な処理は、不連続アクセスを受け付けた場合と同様に、ブロック判定部221でステップS603を実行し、その後は遅延読み出し処理部225及び遅延書き込み処理部224で上述した処理（図7及び図8）を行う。

【0102】そして、保留状態のブロックの書き込み処理が全て終わると、ファイルシステム22は、ファイルオープン処理で確保した遅延書き込み用バッファ223を解放する。また、遅延書き込み処理部224及び遅延読み出し処理部225が、ファイルオープンからファイルクローズのタイミングで実行開始される場合には、それぞれの処理部に終了要求が発行される。なお、ファイルシステム22の起動から終了までの期間、双方の処理部が動作し続ける場合は、終了要求は発行されない。最後に、ファイルシステム22は、ファイルクローズ処理の成功結果をサーバアプリケーション21へ返送し、その処理を終了する。クローズ要求の成功結果は、サーバアプリケーション21からプロトコル処理部25を経由して端末11へ送られる。そして、端末11は、この成

功結果を受け取ることで、ファイルクローズ処理が終了する。

【0103】以上のように、本発明の第1の実施形態に係るファイルシステム22によれば、書き込み要求のデータ範囲をブロックの区切り位置と比較して、ブロック内のデータが揃っているか否かを判定するブロック判定部221を備える。これにより、ブロック内のデータが揃う可能性の少ない状態を特定できる。従って、本当に読み出しの必要なときだけデータ記憶部24からの読み出しを行えるので、常にキャッシュメモリにデータを先読みしておく従来の方式に比べて読み出し処理の無駄がなく、帯域幅の有効利用ができる。さらに、第1の実施形態に係るファイルシステム22では、サーバアプリケーション21から連続的な書き込み要求を受け付けているときは、対応するブロックのデータが遅延書き込み用バッファ223上に存在しなくても（キャッシュにヒットしなくても）、データ記憶部24からデータを読み出さない。そのため、データ記憶部24からの読み出し処理によるサーバアプリケーション21への応答遅延は発生せず、リアルタイムなデータ書き込みが高速に実現できる。一方、不連続な書き込み要求を受け付けたときには、データ記憶部24からデータ読み出しする場合もあるが、この条件では従来の先読み方式でもキャッシュメモリにデータが無いので再読み出しが必要となり、応答遅延の面で同等である。従って、リアルタイムなデータ書き込みが高速に実現できる。

【0104】なお、上記第1の実施形態では、ファイルシステム22が行う動作として、ファイルのオープン処理、クローズ処理及び書き込み処理の3種類のみを説明したが、この他にもファイルの新規作成処理、削除処理及び読み出し処理等の動作を行えるようにしてもよい。

【0105】（第2の実施形態）上記第1の実施形態では、遅延書き込み処理部224からデータ記憶部24へのデータ書き込みを一定の周期で行う場合を説明した。しかし、このデータ書き込みは、ファイル毎に予め定められた周期を用いて行われてもよい。その実現方法の1つとして、各ファイルの拡張属性情報として周期（又は書き込み帯域幅）を所定の場所に記録しておき、ファイルオープン処理時にその情報を読み出して周期を決定する方法がある。情報を記録する場所としては、ファイル名の一部分やファイル先頭部分等のファイル内のデータ領域、又はファイルシステム22がファイルサイズ等の属性情報を管理している領域等が挙げられる。また、他の実現方法の1つとしては、ファイルシステム22に帯域設定用のインタフェースを新たに設け、このインタフェースを使用してサーバアプリケーション21にファイル毎の帯域（周期）を設定してもらう方法がある。以下、第2の実施形態において、この方法を実現するためのファイルシステムを説明する。

【0106】図14は、本発明の第2の実施形態に係る

ファイルシステム27を用いた図1のファイルサーバ20の詳細な構成を示すブロック図である。図14において、ファイルサーバ20は、サーバアプリケーション26と、ファイルシステム27と、ディスクアクセス変換部23と、データ記憶部24と、プロトコル処理部25とを備えている。また、ファイルシステム27は、ブロック判定部271と、バッファ管理部272と、遅延書き込み用バッファ223と、遅延書き込み処理部274と、遅延読み出し処理部225と、帯域設定部276とを備えている。図14に示すように、第2の実施形態に係るファイルシステム27は、基本的には、上記第1の実施形態に係るファイルシステム22に、帯域設定用のインタフェースである帯域設定部276を新たに加えた構成である。以下、ファイルシステム22と異なる部分を中心に、ファイルシステム27を説明する。

【0107】サーバアプリケーション26によって書き込み要求に関する帯域設定の要求（GetResource）が与えられると、帯域設定部276は、指定された帯域幅に基づいて遅延書き込みを行う周期を決定し、遅延書き込み処理部274へ指示する。例えば、帯域幅がRATE [bps]と指定され、遅延書き込み処理部274で1回に書き込む単位をBLKSIZE [バイト]とした場合、周期Tは $(BLKSIZE \times 8) / RATE$  [秒]となる。また、帯域設定部276は、遅延書き込み用バッファ223のサイズを、指定された帯域に基づいて決定してもよい。具体的には、帯域=RATE [bps]（周期=T [秒]）のスピードで書き込み要求を受け付けるファイルは、以下のサイズの遅延書き込み用バッファ223を用意すれば充分である。

バッファ最大サイズ [バイト] =  $BLKSIZE \times ((TwDelay + ToutDelay) / T) + \alpha$

ただし、 $[x]$ は、 $x$ の小数点以下を切り上げた整数を、 $TwDelay$ は、ブロックが確定状態と判定された時点から、遅延書き込み処理部274がディスクアクセス変換部23へライト要求を発行するまでの揺らぎ時間 [秒]を、 $ToutDelay$ は、遅延書き込み処理部274がディスクアクセス変換部23へライト要求を発行してから、実際に書き込みが完了し、遅延書き込み用バッファ223が空くまでの遅延時間 [秒]を、 $\alpha$ は、処理時間の余裕を考慮するための変数（1以上の任意の整数）を表す。

【0108】従って、ファイルオープン時にバッファ管理部272によって確保する遅延書き込み用バッファ223のサイズを、上式で求められるバッファ最大サイズとする。また、サーバアプリケーション26から書き込み要求を受け付け、要求されたデータをブロック判定部271から遅延書き込み用バッファ223へ転送する際には、ブロック判定部271は、ファイルオープン時に確保したバッファ最大サイズを超えてバッファ領域が使用されていないかを調べる。そして、バッファ最大サイ

ズを超えてバッファ領域が使用されていた場合、ブロック判定部 271 は、遅延書き込み処理部 274 で行われる書き込み処理によって遅延書き込み用バッファ 223 が空くのを待つようにする。

【0109】以上のように、本発明の第 2 の実施形態に係るファイルシステム 27 によれば、遅延書き込み用バッファ 223 として用意すべきメモリのサイズが最小限に押さえられる。さらに、サーバアプリケーション 26 が指定帯域を超えて速いスピードで書き込み要求を発行してきたときにも、遅延書き込み用バッファ 223 の空き状況を管理するだけで、メモリ領域を無駄に使用することなく一定の帯域での書き込みを保証できる。なお、上記第 2 の実施形態では、帯域設定部 276 は、帯域設定の要求をサーバアプリケーション 26 から受ける場合を説明したが、別途有するファイル等から帯域設定に関する情報を自ら読み出すようにしてもよい。この場合、ファイルオープン処理がサーバアプリケーション 26 から要求されたタイミングで、帯域設定部 276 が、自立的に帯域設定に関する情報を読み出せばよい。

【0110】（第 3 の実施形態）また、上記第 1 及び第 2 の実施形態では、ファイルサーバ 20 が単一のコンピュータ装置で実現されていることを想定し、図 2 及び図 14 のディスクアクセス変換部 23 もデータ記憶部 24 を制御するドライバを想定して説明したが、ファイルサーバ 20 の構成はこれに限定されない。例えば、図 15 に示す第 3 の実施形態のように、ファイルサーバ 20 が複数のコンピュータ装置（PC）をサーバ内ネットワークで接続した形態であってもよい。

【0111】図 15 は、上記第 1 及び第 2 の実施形態に係るファイルシステムを用いた本発明の第 3 の実施形態に係るクライアント・サーバ・システムの構成を示すブロック図である。図 15 の例では、各端末からのファイルアクセスを受け付けるための複数のファイル入出力用 PC と、各ファイル入出力用 PC からのデータ読み書き要求を受けて、記憶装置とのアクセスを行う複数のディスクアクセス用 PC が、サーバ内ネットワークでそれぞれ接続されている。ファイル入出力用 PC は、それぞれ分散型のファイルシステムを実装し、複数のファイルシステムが連携して、1 つのボリュームを管理する。以上の構成において、ファイル入出力用 PC のファイルシステムに、上述した本発明のファイルシステムを実装し、リアルタイム書き込みを実現させる。なお、この構成の場合、図 2 のディスクアクセス変換部 23 は、図 15 に示すように、サーバ内ネットワーク、各 PC のネットワーク I/F カード、各 PC のサーバ内プロトコル処理部、及び各ディスクアクセス用 PC のディスクドライバの集合体に相当する。また、図 2 のデータ記憶部 24 は、各ディスクアクセス用 PC に対応する記憶装置の集合体に相当する。

【0112】（第 4 の実施形態）また、上記第 1 及び第

2 の実施形態では、クライアント・サーバ・システムのファイルサーバ 20 上に実装されるファイルシステム 22 及び 27 について説明した（ローカル・ファイルシステム）。しかし、図 16 に示す第 4 の実施形態のように、本発明のファイルシステム 22 及び 27 を端末側に実装してもよい（リモート・ファイルシステム）。

【0113】例えば、図 16 に示すような端末 161 に実装するファイルシステム 163 を考える。この場合、通常、ファイルシステム 163 は、クライアントアプリケーション 162 からの書き込み要求をそのままファイルサーバ 166 へ要求するのではなく、端末 161 に設けられたキャッシュメモリ（図示せず）を用いて、ファイルサーバ 166 からの読み出しと、ブロックに相当するサイズによるファイルサーバ 166 への遅延書き込みとを行っている。図 2 及び図 14 と図 16 とで構成を比較すると、図 16 のプロトコル処理部 164 及びネットワーク I/F カード 165 が図 2 及び図 14 のディスクアクセス変換部 23 に相当し、図 16 のファイルサーバ 166 が図 2 及び図 14 のデータ記憶部 24 に相当することとなる。以下、本発明の本発明のファイルシステムを、ローカル・ファイルシステムとして適用した場合と、リモート・ファイルシステムとして適用した場合とで異なる内容を説明する。

【0114】（1）ファイルシステムが出力する書き込み要求の形式が異なるローカル・ファイルシステムである場合（図 2 及び図 14）、ファイルのデータが記憶されているデータ記憶部 24 上の物理的なアドレス情報（データの格納情報）が管理されている。このため、ファイルシステム 22 及び 27 は、サーバアプリケーション 21 が指定したファイルの論理的なオフセット位置に基づいて、データ記憶部 24 上の物理的なアドレスを割り出す。そして、ファイルシステム 22 及び 27 は、ディスクアクセス変換部 23 へ書き込み要求を発行する際には、データを書き込むべきデータ記憶部 24 上の物理的なアドレスを指定する。一方、リモート・ファイルシステムである場合（図 16）、データ記憶部 168 がファイルサーバ 166 上にあるため、その物理的なアドレス情報（データの格納情報）を管理するのは、ファイルサーバ 166 内のファイルシステム（図示せず）の役割である。従って、ファイルシステム 163 は、プロトコル処理部 164 及びネットワーク I/F カード 165 を介して、ファイルサーバ 166 へ書き込み要求を発行する際には、データを書き込むべきデータ記憶部 168 上のアドレスとして、ファイル先頭からの論理的なオフセット位置を指定する。

【0115】（2）リモート・ファイルシステムでは、端数部分の読み出しが不要となる場合があるファイルサーバ 166 側のファイルシステムとしては、書き込み要求を任意のサイズで受け付け可能なものと、固定サイズでしか受け付けられないものの 2 種類がある。リモート

・ファイルシステムにおいてファイルサーバ166側のファイルシステムが前者である場合、ファイルシステム163からプロトコル処理部164及びネットワークI/Fカード165へ発行する書き込み要求も、任意のサイズで行える。従って、この場合、バッファに確保されたブロックの端数部分の読み出しも必要なくなる。以下、リモート・ファイルシステムにおいて、端数部分の読み出しを行わない場合の処理を図17及び図18に示す。図17及び図18は、このリモート・ファイルシステムがクライアントアプリケーションから書き込み要求を受け付けた場合、ブロック判定部が行う動作を示すフローチャートである。なお、この端数部分の読み出しを行わないリモート・ファイルシステムの構成は、図2又は図14から遅延読み出し処理部225を削除した構成に相当する。

【0116】端数部分の読み出しを行わないリモート・ファイルシステムにおける書き込み要求受付処理では、書き込みのサイズが固定しないので、各ブロックでデータの有効な範囲だけが書き込まれる。従って、各ブロックでデータ記憶部に書き込むべき有効な範囲が管理・設定される。図17では、書き込みの開始位置をWriteStartPntと、書き込みの終了位置をWriteEndPntと表記している。そして、ステップS1702、S1705、S1708及びS1710の時点で、WriteStartPnt及びWriteEndPntの値を設定・更新している。

【0117】また、端数部分の読み出しを行っていたステップS502（図6）に代えて、ステップS1702（図18）が実行される。図17のステップS1701で不連続アクセスであると判定された場合には、前回の書き込み要求のPreEndPntが、ブロックの区切り位置か否かが判定される（ステップS1801）。この判定は、PreEndPntがBLKSIZEの倍数であるか否かで行われる。ステップS1801で区切り位置であると判定された場合には、保留状態のブロックが無いと判断され、ステップS1803へ処理が移行する。一方、ステップS1801で区切り位置でないと判定された場合には、保留状態のブロックを確定状態に移させる（ステップS1802）。このとき、ブロック内の有効範囲（データ記憶部に書き込むべき範囲）の終了位置WriteEndPntを、PreEndPntに応じた値に設定する。その後、遅延書き込み用バッファとして1ブロック分の領域が新たに確保され（ステップS1803）、そのブロックが保留状態に初期設定される（ステップS1804）。そして、端数部分の読み出しを行わないリモート・ファイルシステムにおける遅延書き込み処理では、図8のステップS804のみ異なり、上記有効範囲の情報（WriteStartPnt及びWriteEndPnt）に基づいて、ディスクアクセス変換部へのライト要求が発行される。

【0118】上記のように、端末側にあるファイルシステムに本発明の方式を実装すれば、データ読み出し回数及びデータサイズを圧縮でき、端末とファイルサーバとの間の帯域幅を有効利用できる。さらに、リモートファイルのアクセスであっても、サーバアプリケーションの応答時間を短縮でき、リアルタイムでの高速書き込みが可能になる。

【0119】なお、上記各実施形態では、保留状態と判定されたブロックのデータは、それ以降に書き込み要求を受け付けられない限り、遅延書き込み用バッファ223上に保持され続け、最終的にはファイルクローズ要求を受け付けたときにデータ記憶部24へ書き込まれる。しかしながら、記憶装置に保存すべきデータを、長期間メモリ上に置いておくと、システムがクラッシュしてデータが消えてしまうような事故に遭う可能性が高くなる。従って、ブロックが保留状態と判定されてからの経過時間を測定する構成をさらに設け、一定時間経過しても書き込み要求を受け付けなかった場合には、タイムアウトしたものとみなし、ファイルクローズ要求の受け付け時に保留状態のブロックが残っていた場合と同様の処理を行うようにしてもよい。さらに、上記第2の実施形態のように、帯域設定用インタフェース（帯域設定部276）を設けた場合には、タイムアウトを検出する時間を指定された帯域に基づいて決定することもできる。

【0120】また、上記各実施形態では、データ記憶部24に対して常にブロックサイズでの書き込みを行っていたが、必要に応じて1回の書き込みサイズを変更してもよい。例えば、ブロック判定部221で行う不連続アクセスの読み出し処理（ステップS502）において、データ無効範囲が発生した場合、常に読み出し要求を発行していたが、データ無効範囲が最小アクセス単位であれば、読み出し要求せずに、ブロック内の有効なデータ範囲のみデータ記憶部24へ書き込むようにしてもよい。実現方法としては、図6のステップS603及びS605の前に、データ無効範囲が最小アクセス単位であるか否かを判定する。最小アクセス単位である場合には、遅延書き込み処理部224によって書き込むべきデータの範囲が特定できる情報を渡し、ブロックを確定状態に設定する。遅延書き込み処理部224では、ブロック内の指定された範囲のみデータ記憶部24へ書き込む。最小アクセス単位でない場合には、図6で説明したのと同じ方法で、データの読み出しを行う。この方法では、データ読み出し回数及びデータサイズをさらに圧縮でき、さらなる帯域幅の有効利用が図れる。

#### 【図面の簡単な説明】

【図1】本発明によって提供されるファイルシステムが適用されるクライアント・サーバ・システムの構成例を示すブロック図である。

【図2】本発明の第1の実施形態に係るファイルシステム22を用いた図1のファイルサーバ20の詳細な構成

【図3】処理される各データの単位及びサイズの関係を示す図である。

【図５】ブロック判定部２２１がサーバアプリケーション２１から書き込み要求を受け付けた場合に行う動作を示すフローチャートである。

【図 7】遅延書き込み処理部 224 が行う動作を示すフローチャートである。

【図 9】データ記憶部 24 から読み出されたデータが遅延書き込み用バッファ 223 に格納される方法の一例を示す図である。

【図１０】書き込み要求のデータ範囲とブロックの区切り位置との関係の一例を示す図である（連続アクセス）。

【図 11】書き込み要求のデータ範囲とブロックの区切り位置との関係の一例を示す図である（不連続アクセス）。

【図 1 2】書き込み要求のデータ範囲とブロックの区切り位置との関係の一例を示す図である（不連続アクセス）。

【図 1 3】書き込み要求のデータ範囲とブロックの区切り位置との関係の一例を示す図である（不連続アクセス）。

【図 14】 本発明の第 2 の実施形態に係るファイルシス

【図１５】第１及び第２の実施形態に係るファイルシステムを用いた本発明の第３の実施形態に係るクライアント・サーバ・システムの構成を示すブロック図である。

【図１６】第１及び第２の実施形態に係るファイルシステムを用いた本発明の第４の実施形態に係るクライアント・サーバ・システムの構成を示すブロック図である。

【図17】ファイルシステム163がクライアントアプリケーション162から書き込み要求を受け付けた場合に行う動作を示すフローチャートである。

【図18】ファイルシステム163がクライアントアプリケーション162から書き込み要求を受け付けた場合に行う動作を示すフローチャートである。

1 1 ~ 1 N, 1 6 1 … 端末

2 0, 1 6 6 … ファイルサーバ

2 1, 2 6, 1 6 7 … サーバアプリケーション

2 2, 2 7, 1 6 3 … ファイルシステム

2 3 … ディスクアクセス変換部

2 4, 1 6 8 … データ記憶部

2 5, 1 6 4 … プロトコル処理部

3 0 … ネットワーク

1 6 2 … クライアントアプリケーション

1 6 5 … ネットワーク I / F カード

2 2 1, 2 7 1 … ブロック判定部

2 2 2, 2 7 2 … バッファ管理部

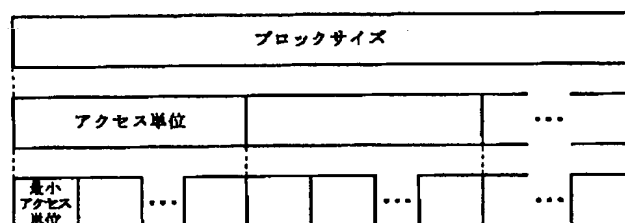
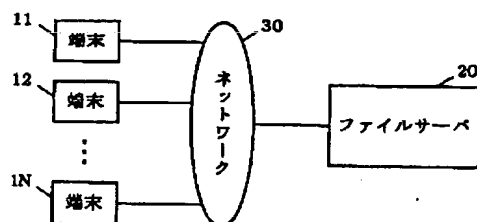
2 2 3 … 遅延書き込み用バッファ

2 2 4, 2 7 4 … 遅延書き込み処理部

2 2 5 … 遅延読み出し処理部

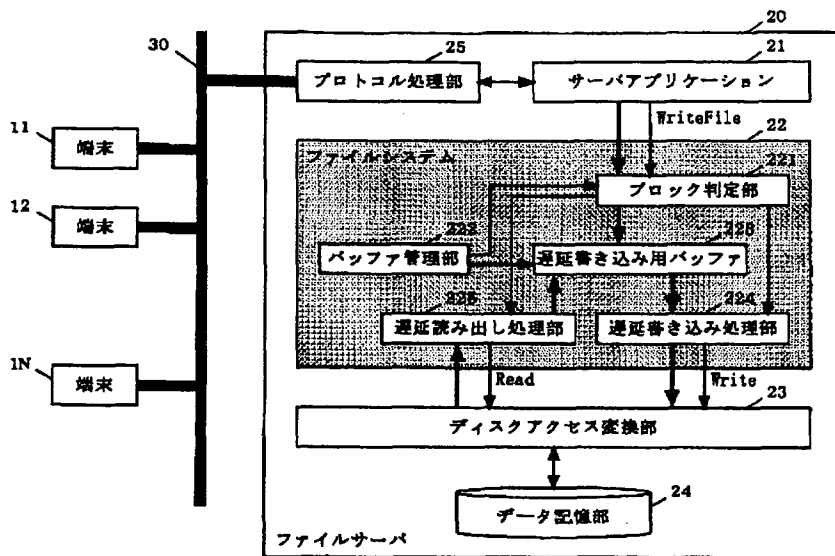
2 7 6 … 帯域設定部

【图 3】

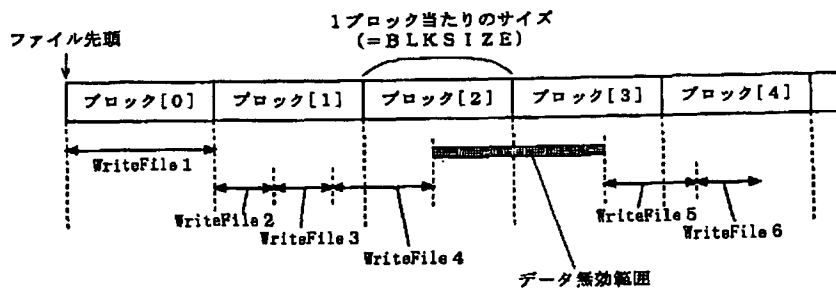




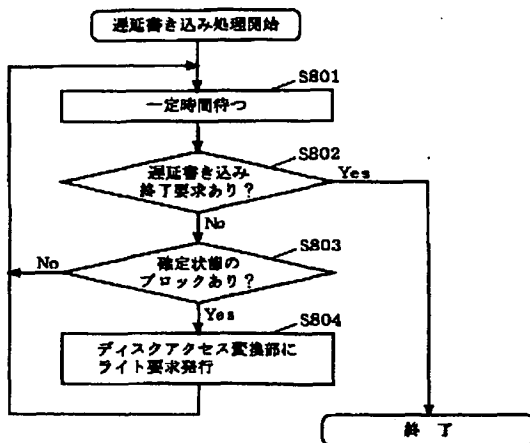
【図2】



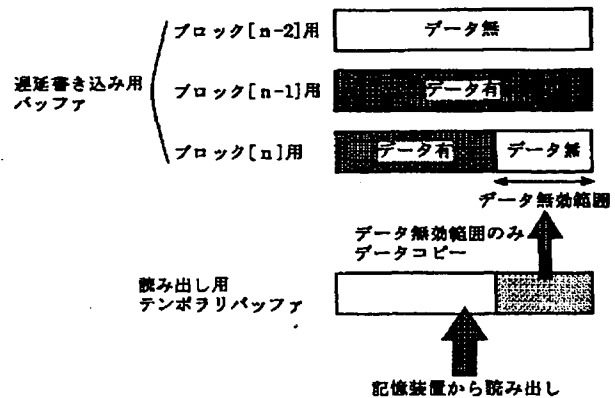
【図4】



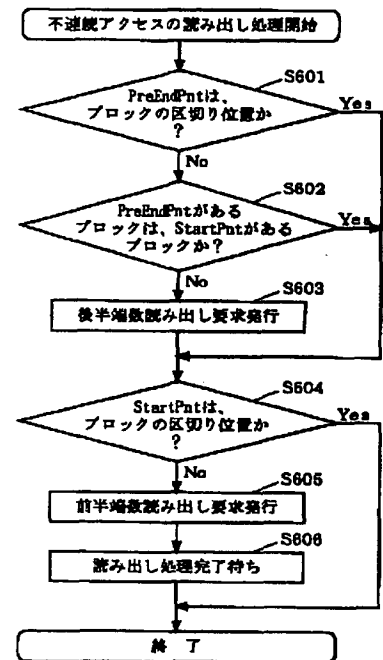
【図8】



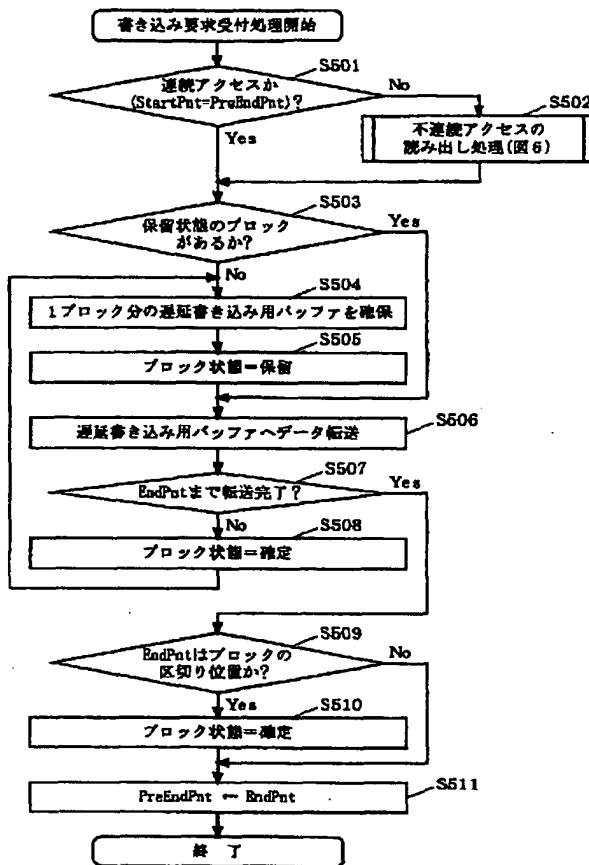
【図9】



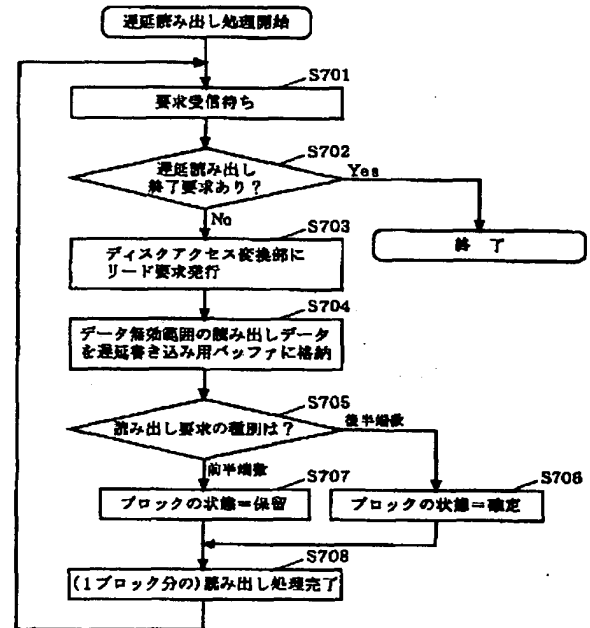
【図6】



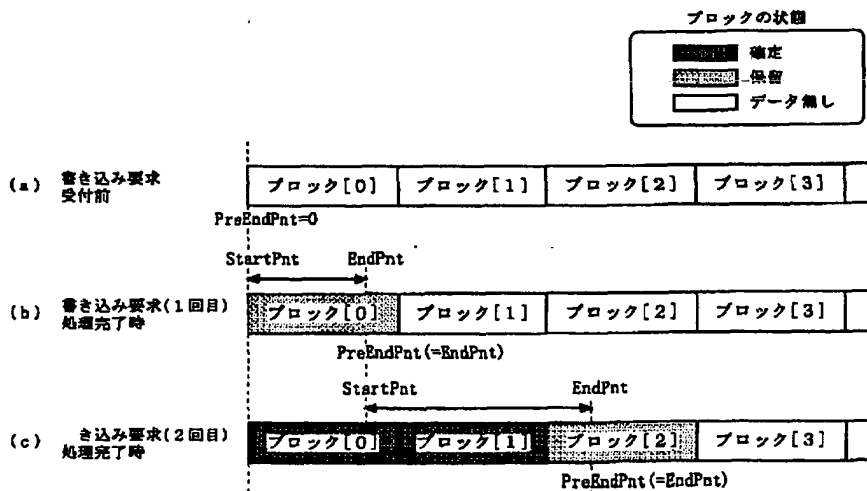
【図5】



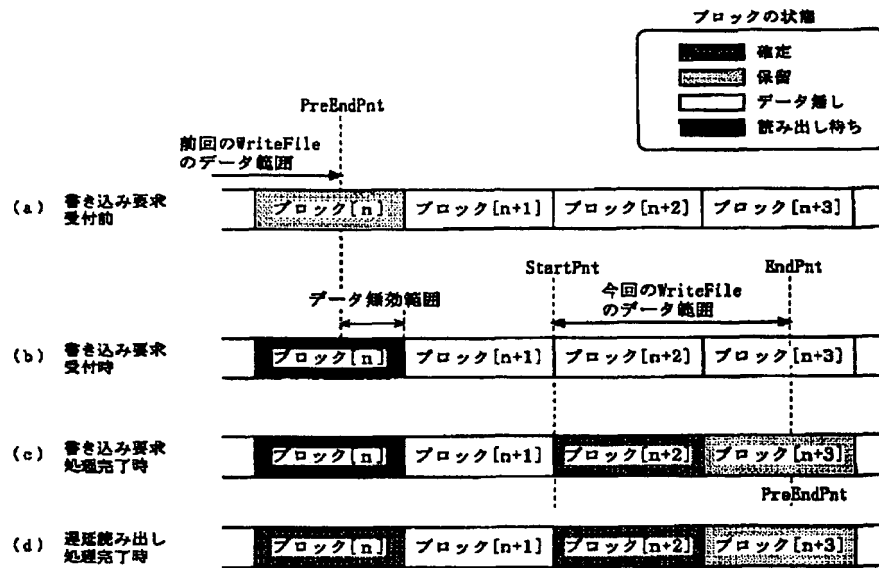
【図7】



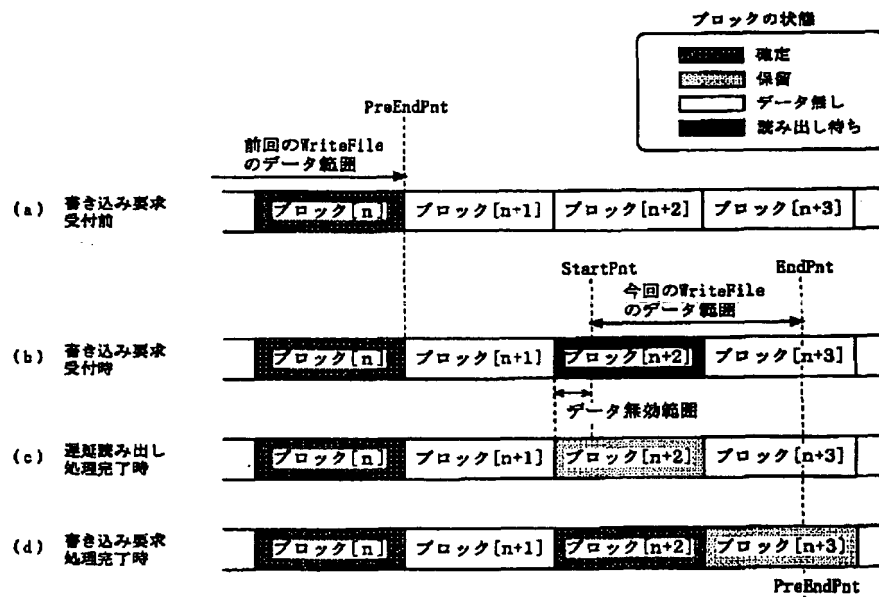
【図10】



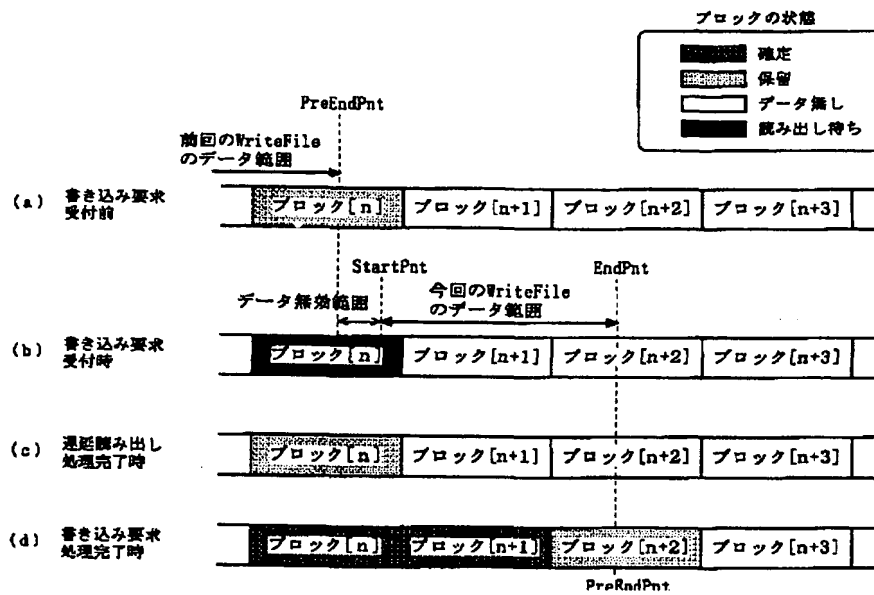
【図11】



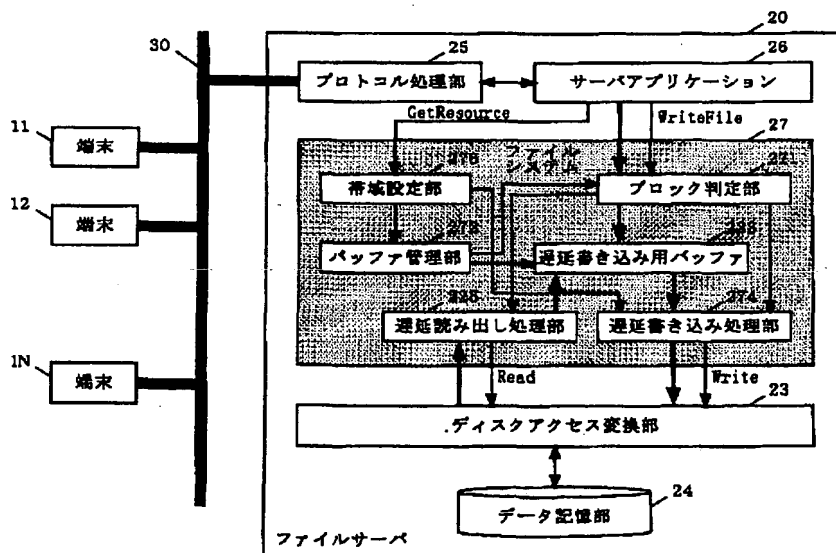
【図12】



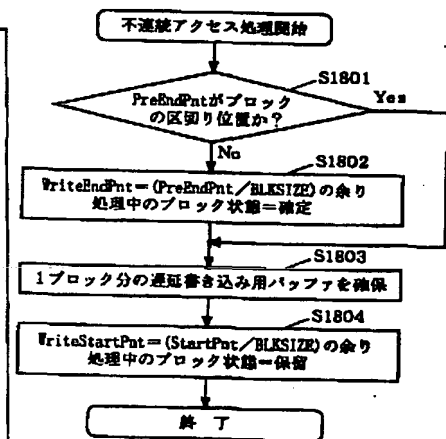
【図13】



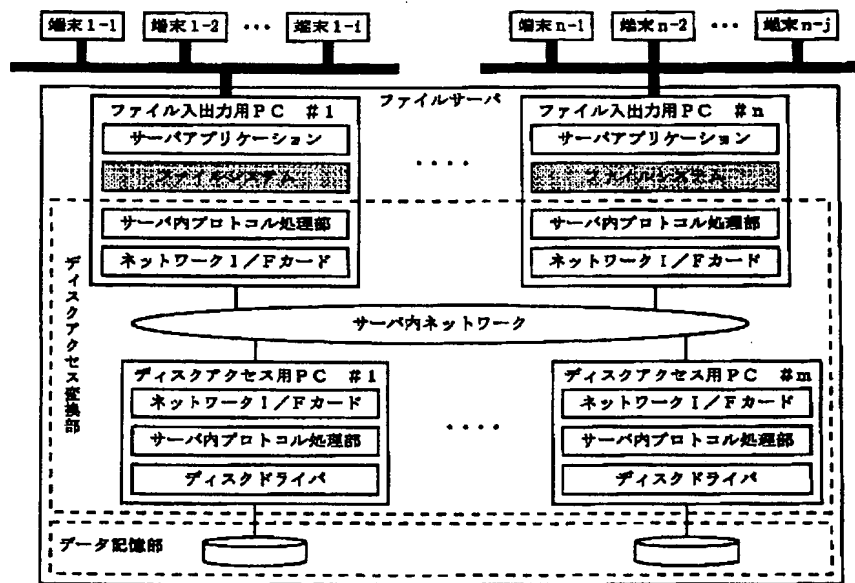
【図14】



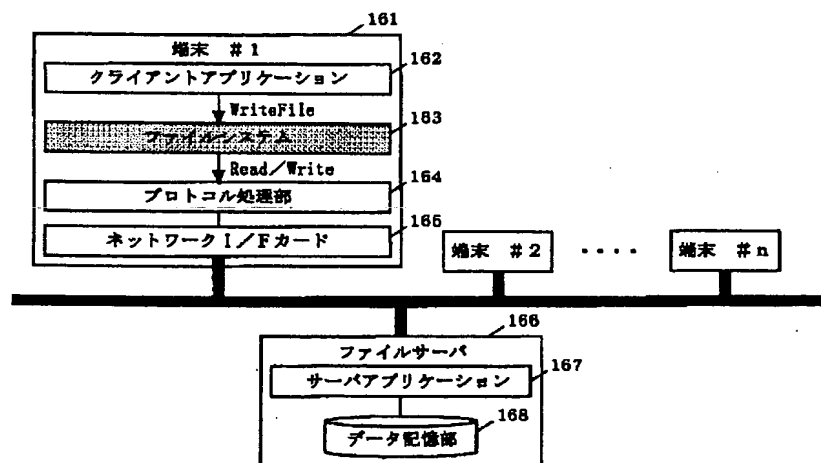
【図18】



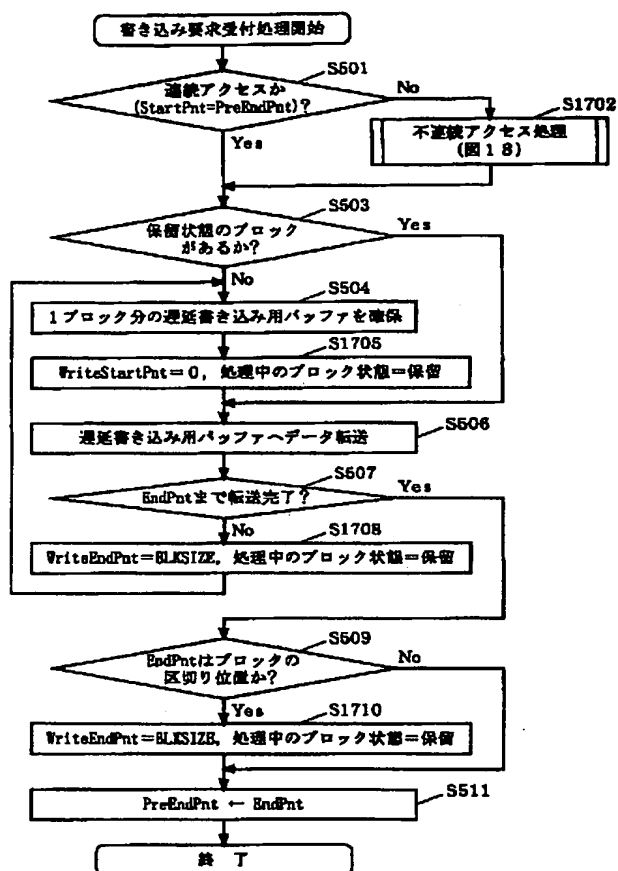
【図 15】



【図 16】



【図17】



フロントページの続き

(72)発明者 辻 敦宏  
大阪府門真市大字門真1006番地 松下電器  
産業株式会社内

Fターム(参考) 5B082 FA02 FA03 HA05